

Tilburg University

On the Convexity of Step out - Step in Sequencing Games

Musegaas, Marieke; Borm, Peter; Quant, Marieke

Publication date:
2016

Document Version
Early version, also known as pre-print

[Link to publication in Tilburg University Research Portal](#)

Citation for published version (APA):
Musegaas, M., Borm, P., & Quant, M. (2016). *On the Convexity of Step out - Step in Sequencing Games*. (CentER Discussion Paper; Vol. 2016-043). CentER, Center for Economic Research.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

No. 2016-043

**ON THE CONVEXITY OF STEP OUT – STEP IN SEQUENCING
GAMES**

By

Marieke Musegaas, Peter Borm, Marieke Quant

17 November 2016

ISSN 0924-7815
ISSN 2213-9532

On the Convexity of Step out - Step in Sequencing Games

M. Musegaas^{*†} P.E.M. Borm^{*} M. Quant^{*}

November 17, 2016

Abstract

The main result of this paper is the convexity of Step out - Step in (SoSi) sequencing games, a class of relaxed sequencing games first analyzed by Musegaas, Borm, and Quant (2015). The proof makes use of a polynomial time algorithm determining the value and an optimal processing order for an arbitrary coalition in a SoSi sequencing game. In particular, we use that in determining an optimal processing order of a coalition $S \cup \{i\}$, the algorithm can start from the optimal processing order found for coalition S and thus all information on this optimal processing order of S can be used.

Keywords: relaxed sequencing games, convexity

JEL classification number: C71, C44

1 Introduction

This paper considers one-machine sequencing situations in which a certain number of players, each with one job, have to be served by a single machine. The processing time of a job is the time the machine takes to process the corresponding job of this player. Every player has an individual linear cost function, specified by an individual cost parameter, which depends on the completion time of this job which is defined as the sum of the processing times of his own job and the jobs that are processed before his own job. There are no further restrictive assumptions such as due dates, ready times or precedence constraints imposed on the jobs. Smith (1956) showed that the total joint costs are minimal if the jobs are processed in non-increasing order with respect to their urgency, defined as the ratio of the individual cost parameter and the processing time.

We assume that the players are arranged in an initial order and thus the rearrangement of the initial order to an optimal order will lead to cost savings. To analyze the problem on how these cost savings should be allocated to the players, sequencing games are introduced. The value of a coalition in a sequencing game serves as a benchmark for determining a fair allocation of the optimal cost savings and represents the “virtual” maximal cost savings which this coalition can achieve by means of admissible rearrangements.

^{*}CentER and Department of Econometrics and Operations Research Tilburg University, P.O. Box 90153, 5000 LE Tilburg, The Netherlands

[†]Correspondence to: m.musegaas@tilburguniversity.edu

Which rearrangements are admissible for a coalition is a modelling choice. The classical assumption made in Curiel, Pederzoli, and Tijs (1989) is that two players of a certain coalition can only swap their positions if all players between them are also members of the coalition. They show that the resulting sequencing games are convex and therefore have a non-empty core. Relaxed sequencing games arise by relaxing this classical assumption about the set of admissible rearrangements for coalitions in a consistent way.

In Curiel, Potters, Prasad, Tijs, and Veltman (1993) four different relaxed sequencing games are introduced. These relaxations are based on requirements for the players outside the coalition regarding either their position in the processing order or their starting time. Slikker (2006) considered these four relaxed sequencing games in more detail by investigating the corresponding cores. In Van Velzen and Hamers (2003) two further classes of relaxed sequencing games are considered. In relaxed sequencing games the values of coalitions become larger because the set of admissible rearrangements is larger than in the classical case. As a consequence, while classical sequencing games are convex, relaxed sequencing games might not be convex anymore. To the best of our knowledge there is no general convexity result with respect to specific subclasses of relaxed sequencing games.

In Musegaas et al. (2015) an alternative class of relaxed sequencing games is considered, the class of Step out - Step in (SoSi) sequencing games. In a SoSi sequencing game a member of a coalition is allowed to step out from his position in the processing order and to step in at any position later in the processing order. Providing an upper bound on the values of the coalitions in a SoSi sequencing game, Musegaas et al. (2015) showed that every SoSi sequencing game has a non-empty core. In Musegaas et al. (2015) also a polynomial time algorithm is provided to determine the value and an optimal processing order for an arbitrary coalition in a SoSi sequencing game. This paper shows, by means of this polynomial time algorithm, that SoSi sequencing games are convex. For proving this, we use a specific feature of the algorithm. Namely, for determining an optimal processing order for a coalition, one can use the information of the optimal processing orders of subcoalitions. More precisely, if one wants to know an optimal processing order for a coalition $S \cup \{i\}$, then the algorithm can start from the optimal processing order found for coalition S . In particular, this helps to analyze the marginal contribution of a player i to joining coalitions S, T with $S \subseteq T$ and $i \notin T$, and thus it helps to prove the convexity of SoSi sequencing games.

The organization of this paper is as follows. Section 2 recalls basic definitions on one-machine sequencing situations and the formal definition of a SoSi sequencing game. Section 3 identifies a number of important key features of the algorithm of Musegaas et al. (2015) that are especially useful in proving the convexity of SoSi sequencing games. In Section 4 the proof of convexity for SoSi sequencing games is provided.

2 SoSi sequencing games

This section recalls basic definitions on one-machine sequencing situations and related SoSi sequencing games.

A one-machine sequencing situation can be summarized by a tuple (N, σ_0, p, α) , where N is the set of players, each with one job to be processed on the single machine. A processing order of the players can be described by a bijection $\sigma : N \rightarrow \{1, \dots, |N|\}$. More specifically, $\sigma(i) = k$ means that player i is in position k . Let $\Pi(N)$ denote the set of all such processing orders. The processing order $\sigma_0 \in \Pi(N)$ specifies the initial order. The processing time $p_i > 0$ of the job of player i is the time the machine takes to process

this job. The vector $p \in \mathbb{R}_{++}^N$ summarizes the processing times. Furthermore, the costs of player i of spending t time units in the system is assumed to be determined by a linear cost function $c_i : [0, \infty) \rightarrow \mathbb{R}$ given by $c_i(t) = \alpha_i t$ with $\alpha_i > 0$. The vector $\alpha \in \mathbb{R}_{++}^N$ summarizes the coefficients of the linear cost functions. It is assumed that the machine starts processing at time $t = 0$, and also that all jobs enter the system at $t = 0$.

The total joint costs of a processing order $\sigma \in \Pi(N)$ are given by $\sum_{i \in N} \alpha_i C_i(\sigma)$, where $C_i(\sigma)$ denotes the completion time of player i and is defined by

$$C_i(\sigma) = \sum_{j \in N: \sigma(j) \leq \sigma(i)} p_j,$$

A processing order is called optimal if it minimizes the total joint costs over all possible processing orders. In Smith (1956) it is shown that in each optimal order the players are processed in non-increasing order with respect to their *urgency* u_i defined by $u_i = \frac{\alpha_i}{p_i}$. The maximal total cost savings are equal to the difference in total costs between the initial order and an optimal order.

A *coalitional game* is a pair (N, v) where N denotes a non-empty, finite set of players and $v : 2^N \rightarrow \mathbb{R}$ assigns a monetary payoff to each coalition $S \in 2^N$, where 2^N denotes the collection of all subsets of N . In general, the value $v(S)$ equals the highest payoff the coalition S can jointly generate by means of optimal cooperation without help of players in $N \setminus S$. By convention, $v(\emptyset) = 0$.

To tackle the allocation problem of the maximal cost savings in a sequencing situation (N, σ_0, p, α) , one can analyze an associated coalitional game (N, v) . Here N naturally corresponds to the set of players in the game and, for a coalition $S \subseteq N$, $v(S)$ reflects the maximal cost savings this coalition can make with respect to the initial order σ_0 . In order to determine these maximal cost savings, assumptions must be made on the possible reorderings of coalition S with respect to the initial order σ_0 .

The classical (strong) assumption is that a member of a certain coalition $S \subset N$ can only swap with another member of the coalition if all players between these two players, according to the initial order, are also members of S . Note that the resulting set of admissible reorderings for a coalition is quite restrictive, because there may be more reorderings possible which do not hurt the interests of the players outside the coalition.

In a SoSi sequencing game a member of the coalition S is allowed to step out from his position in the processing order and to step in at any position later in the processing order. Note that from an optimality point of view it is clear that one can assume without loss of generality that a member of S who steps out, only steps in at a position directly behind another member of the coalition S . This means that for every players outside S its set of predecessors can not become larger and its direct follower was already a follower of him in the initial order. Hence, a processing order σ is called *admissible* for S in a SoSi sequencing game if

- (i) $P(\sigma, i) \subseteq P(\sigma_0, i)$ for all $i \in N \setminus S$,
- (ii) $\sigma^{-1}(\sigma(i) + 1) \in F(\sigma_0, i)$ for all $i \in N \setminus S$ with $\sigma(i) \neq |N|$,

where $P(\sigma, i) = \{j \in N \mid \sigma(j) < \sigma(i)\}$ denotes the set of predecessors of player i with respect to processing order σ and $F(\sigma, i) = \{j \in N \mid \sigma(j) > \sigma(i)\}$ denotes the set of followers. Given an initial order σ_0 the set of admissible orders for coalition S is denoted

by $\mathcal{A}(\sigma_0, S)$. Correspondingly, Musegaas et al. (2015) defined the *Step out, Step in (SoSi) sequencing game* (N, v) by

$$v(S) = \max_{\sigma \in \mathcal{A}(\sigma_0, S)} \sum_{i \in S} \alpha_i (C_i(\sigma_0) - C_i(\sigma)),$$

for all $S \subseteq N$. A processing order $\sigma^* \in \mathcal{A}(\sigma_0, S)$ is called *optimal for S* if

$$\sum_{i \in S} \alpha_i (C_i(\sigma_0) - C_i(\sigma^*)) = \max_{\sigma \in \mathcal{A}(\sigma_0, S)} \sum_{i \in S} \alpha_i (C_i(\sigma_0) - C_i(\sigma)).$$

Note that a processing order is admissible for a coalition in a classical sequencing game if there is an equality in condition (i). Therefore, given a coalition, the corresponding set of admissible orders in a SoSi sequencing game is larger than the set of admissible orders in the corresponding classical sequencing game. As a consequence, the values of coalitions in SoSi sequencing games can become larger with respect to classical sequencing games.

The following example provides an instance of a SoSi sequencing game.

Example 2.1. Consider a one-machine sequencing situation with $N = \{1, 2, 3\}$. The vector of processing times is $p = (3, 2, 1)$, the vector of coefficients corresponding to the linear cost functions is $\alpha = (4, 6, 5)$ and the initial order is $\sigma_0 = (1 \ 2 \ 3)$. Let (N, v) be the corresponding SoSi sequencing game. Table 1 provides the values of all coalitions.

S	$\{1\}$	$\{2\}$	$\{3\}$	$\{1, 2\}$	$\{1, 3\}$	$\{2, 3\}$	N
$v(S)$	0	0	0	10	3	4	25

Table 1: The SoSi sequencing game of Example 2.1

Note that the values of the coalitions in the game (N, v) are equal to the values of the coalitions in the classical sequencing game of this one-machine sequencing situation except for the only disconnected coalition, coalition $\{1, 3\}$. Coalition $\{1, 3\}$ cannot save costs in the classical sequencing game because there exists no admissible order other than the initial order. However, in the SoSi sequencing game coalition $\{1, 3\}$ has two admissible orders:

$$\mathcal{A}(\sigma_0, \{1, 3\}) = \{(1 \ 2 \ 3), (2 \ 3 \ 1)\}^1$$

These processing orders are illustrated in Figure 1. Hence, the value of coalition $\{1, 3\}$ is given by

$$v(\{1, 3\}) = \max \left\{ 0, \sum_{i \in \{1, 3\}} \alpha_i (C_i((1 \ 2 \ 3)) - C_i((2 \ 3 \ 1))) \right\} = \max\{0, -12 + 15\} = 3. \quad \triangle$$

¹Processing order $(2 \ 3 \ 1)$ means that player 2 is in the first position, player 3 in the second position and player 1 in the last position.

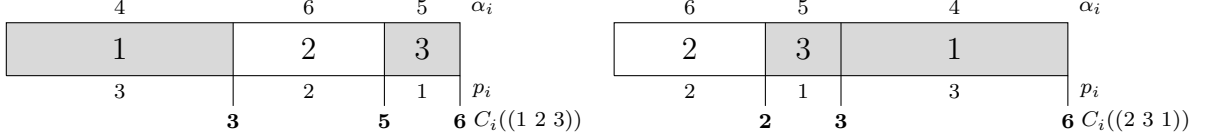


Figure 1: The two admissible orders for coalition $\{1, 3\}$ in Example 2.1

3 On the algorithm for finding the values of the coalitions

Musegaas et al. (2015) provided a polynomial time algorithm to determine an optimal order for every possible coalition and, consequently, the values of the coalitions. For proving convexity of SoSi sequencing games, we use specific key features of this algorithm. In this section we will derive and summarize these specific features. For example, in Theorem 3.4, we will show that in determining an optimal processing order of a coalition $S \cup \{i\}$ in a SoSi sequencing game, the algorithm can start from the optimal processing order found for coalition S .

We start with recalling some definitions such as components. For $S \in 2^N \setminus \{\emptyset\}$, $\sigma \in \Pi(N)$ and $s, t \in N$ with $\sigma(s) < \sigma(t)$, define

$$\begin{aligned} S^\sigma(s, t) &= \{i \in S \mid \sigma(s) < \sigma(i) < \sigma(t)\}, \\ \bar{S}^\sigma(s, t) &= \{i \in N \setminus S \mid \sigma(s) < \sigma(i) < \sigma(t)\}, \\ S^\sigma[s, t] &= \{i \in S \mid \sigma(s) \leq \sigma(i) \leq \sigma(t)\}, \\ \bar{S}^\sigma[s, t] &= \{i \in N \setminus S \mid \sigma(s) \leq \sigma(i) \leq \sigma(t)\}. \end{aligned}$$

The sets of players $S^\sigma[s, t]$, $\bar{S}^\sigma[s, t]$, $S^\sigma(s, t)$ and $\bar{S}^\sigma(s, t)$ are defined in a similar way.

A coalition $S \in 2^N \setminus \{\emptyset\}$ is called *connected* with respect to σ_0 if for all $i, j \in S$ and $k \in N$ such that $\sigma_0(i) < \sigma_0(k) < \sigma_0(j)$ it holds that $k \in S$. A connected coalition $U \subseteq S$ with respect to σ_0 is called a *component* of S with respect to σ_0 if $U \subseteq U' \subseteq S$ and U' connected with respect to σ_0 implies that $U' = U$. Let $h(\sigma_0, S) \geq 1$ denote the number of components of S with respect to σ_0 . The partition of S into components with respect to σ_0 is denoted by

$$S \setminus \sigma_0 = \left\{ S_1^{\sigma_0}, S_2^{\sigma_0}, \dots, S_{h(\sigma_0, S)}^{\sigma_0} \right\},$$

with for each $k \in \{1, \dots, h(\sigma_0, S) - 1\}$, $i \in S_k^{\sigma_0}$ and $j \in S_{k+1}^{\sigma_0}$ we have $\sigma_0(i) < \sigma_0(j)$. In the same way processing order σ_0 divides $N \setminus S$ into subgroups. For this, define

$$\begin{aligned} \bar{S}_0^{\sigma_0} &= \{i \in N \setminus S \mid \sigma_0(i) < \sigma_0(j) \text{ for all } j \in S_1^{\sigma_0}\}, \\ \bar{S}_{h(\sigma_0, S)}^{\sigma_0} &= \{i \in N \setminus S \mid \sigma_0(i) > \sigma_0(j) \text{ for all } j \in S_{h(\sigma_0, S)}^{\sigma_0}\}, \\ \bar{S}_k^{\sigma_0} &= \{i \in N \setminus S \mid \sigma_0(j) < \sigma_0(i) < \sigma_0(l) \text{ for all } j \in S_k^{\sigma_0}, \text{ for all } l \in S_{k+1}^{\sigma_0}\}, \end{aligned}$$

for all $k \in \{1, \dots, h(\sigma_0, S) - 1\}$. Notice that $\bar{S}_0^{\sigma_0}$ and $\bar{S}_{h(\sigma_0, S)}^{\sigma_0}$ might be empty sets, but $\bar{S}_k^{\sigma_0} \neq \emptyset$ for all $k \in \{1, \dots, h(\sigma_0, S) - 1\}$. See Figure 2 for an illustration of the subdivision of S and $N \setminus S$ into subgroups by means of processing order σ_0 .

Note that for given $S \subseteq N$ it is possible that a processing order $\sigma \in \mathcal{A}(\sigma_0, S)$ contains less components than σ_0 , because all players of a certain component with respect to S

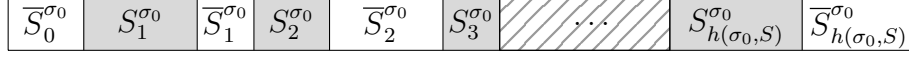


Figure 2: Partition of the players in S and $N \setminus S$ with respect to an order σ_0

may step out from this component and join other components. For $\sigma \in \mathcal{A}(\sigma_0, S)$ with $\sigma_0 \in \Pi(N)$, define *modified components* $S_1^{\sigma_0, \sigma}, \dots, S_{h(\sigma_0, S)}^{\sigma_0, \sigma}$ by

$$S_k^{\sigma_0, \sigma} = \{i \in S \mid \sigma(j) < \sigma(i) < \sigma(l) \text{ for all } j \in \overline{S}_{k-1}^{\sigma_0}, \text{ for all } l \in \overline{S}_k^{\sigma_0}\},$$

for all $k \in \{1, \dots, h(\sigma_0, S)\}$. Hence, $S_k^{\sigma_0, \sigma}$ consists of the group of players that are positioned in processing order σ in between the subgroups $\overline{S}_{k-1}^{\sigma_0}$ and $\overline{S}_k^{\sigma_0}$.

Note that $S_k^{\sigma_0, \sigma}$ might be empty for some k while

$$\bigcup_{k=1}^{h(\sigma_0, S)} S_k^{\sigma_0, \sigma} = S.$$

Moreover, recall that a player is not allowed to move to an earlier component (condition (i) of admissibility) but he is allowed to move to any position later in the processing order and thus we have

$$\bigcup_{k=1}^l S_k^{\sigma_0, \sigma} \subseteq \bigcup_{k=1}^l S_k^{\sigma_0},$$

for all $l \in \{1, \dots, h(\sigma_0, S)\}$. Furthermore, denote the index of the corresponding modified component of player $i \in S$ in processing order σ with respect to initial processing order σ_0 by $c(i, S, \sigma)$, where

$$c(i, S, \sigma) = k \text{ if and only if } i \in S_k^{\sigma_0, \sigma}.$$

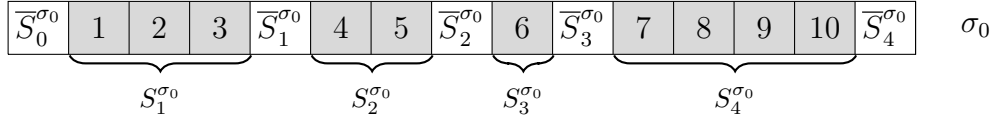
Since the component index of player $i \in S$ with respect to σ can only be increased (due to condition (i) of admissibility), we have

$$c(i, S, \sigma) \geq c(i, S, \sigma_0).$$

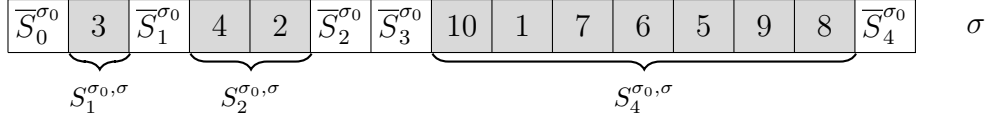
An illustration of the definitions of components, modified components and the index $c(i, S, \sigma)$ can be found in the following example.

Example 3.1. Consider a one-machine sequencing situation (N, σ_0, p, α) with $S \subseteq N$ such that $S = \{1, 2, \dots, 10\}$. In Figure 3(a) an illustration can be found of initial processing order σ_0 and the partition of S into components. Next, consider processing order σ as illustrated in Figure 3(b) that is admissible for S . Note that σ contains less components than σ_0 . Figure 3(b) also illustrates the definition of modified components. Note that there is one modified component that is empty, namely $S_3^{\sigma_0, \sigma}$. Since player 3 belongs to the first modified component, we have $c(3, S, \sigma) = 1$. Moreover, since player 3 is the only player who belongs to the first modified component, we have $S_1^{\sigma_0, \sigma} = \{3\}$. Similarly, we have $c(4, S, \sigma) = c(2, S, \sigma) = 2$, and $c(i, S, \sigma) = 4$, for all $i \in S \setminus \{2, 3, 4\}$. \triangle

To find an optimal order for every possible coalition Musegaas et al. (2015) provided a polynomial time algorithm. Given a one machine sequencing situation (N, σ_0, p, α) and a coalition $S \in 2^N \setminus \{\emptyset\}$, the polynomial time algorithm introduced by Musegaas et al.



(a) Illustration of the components of S with respect to σ_0



(b) Illustration of the modified components of S with respect to σ and initial order σ_0

Figure 3: Illustration of components and modified components

(2015) starts with a preprocessing step. In this preprocessing step the players within the components of S are reordered such that they are in non-increasing order with respect to their urgency. This is done by setting the initial processing order σ_0 equal to the processing order σ_0^S , where $\sigma_0^S \in \mathcal{A}(\sigma_0, S)$ is the unique urgency respecting processing order such that for all $i \in S$

$$c(i, S, \sigma_0^S) = c(i, S, \sigma_0), \quad (1)$$

where a processing order $\sigma \in \Pi(N)$ is called *urgency respecting* with respect to S if

(i) (σ is *componentwise optimal*) for all $i, j \in S$ with $c(i, S, \sigma) = c(j, S, \sigma)$:

$$\sigma(i) < \sigma(j) \Rightarrow u_i \geq u_j.$$

(ii) (σ satisfies *partial tiebreaking*) for all $i, j \in S$ with $c(i, S, \sigma_0) = c(j, S, \sigma_0)$:

$$u_i = u_j, \sigma_0(i) < \sigma_0(j) \Rightarrow \sigma(i) < \sigma(j).$$

Note that (1) states that all players in S stay in their component, i.e., the partition of S into components stays the same. Moreover, condition (i) of urgency respecting states that the players within a component of S are in non-increasing order with respect to their urgency. Moreover, a tiebreaking rule in condition (ii) ensures that if there are two players with the same urgency in the same component of S with respect to σ_0 , then the player who was first in σ_0 is earlier in processing order σ . Note that the partial tiebreaking condition does not imply anything about the relative order of two players with the same urgency who are in the same component of S with respect to σ but who were in different components of S with respect to σ_0 . Therefore, an arbitrary urgency respecting order does not need to be unique, but σ_0^S , because of condition (1), is.

After the preprocessing step, the players in S are considered in reverse order with respect to σ_0^S and for every player the algorithm checks whether moving the player to a certain position later in the processing order is beneficial. If so, then the algorithm will move this player. The algorithm works in a greedy way in the sense that every player is moved to the position giving the highest cost savings at that moment. Moreover, every player is considered in the algorithm exactly once and every player is moved to another position in the processing order at most once. The obtained processing order after the complete run of the algorithm is denoted by σ_S .

The following properties follow directly from the definition and the characteristics of the algorithm to find the optimal processing order σ_S for coalition S and will be used in this paper in order to show that SoSi sequencing games are convex.

- Property (i): After every step during the run of the algorithm, we have a processing order that is urgency respecting with respect to S .
- Property (ii): If during the run of the algorithm a player is moved to a position later in the processing order, then this results in strictly positive cost savings which corresponds to the, at that instance, highest possible cost savings. In case of multiple options, we choose the component that is most to the left and, in that component, we choose the position that is most to the left.
- Property (iii): The mutual order between players who have already been considered will stay the same during the rest of the run of the algorithm.
- Property (iv): The processing order σ_S is the unique optimal processing order such that no player can be moved to an earlier component while the total costs remain the same. Also, if there are two players with the same urgency in the same component, then the player who was first in σ_0 is earlier in processing order σ_S .
- Property (v): If it is admissible with respect to σ_0 to move a player to a component more to the left with respect to order σ_S , then moving this player to this component will lead to higher total costs.

An interesting property for the urgencies of players in an optimal order is that if it is admissible that two players switch position, then the player with the highest urgency should be positioned first. This is stated in the following proposition.

Proposition 3.1 (cf. Lemma 4.1 in Musegaas et al. (2014)). *Let (N, σ_0, p, α) be a one-machine sequencing situation, let $S \in 2^N \setminus \{\emptyset\}$ and let $\sigma \in \mathcal{A}(\sigma_0, S)$ be an optimal order for S . Let $k, l \in S$ with $\sigma(k) < \sigma(l)$ and $c(l, \sigma_0, S) \leq c(k, \sigma, S)$. Then, $u_k \geq u_l$.*

From the previous proposition together with the fact that the algorithm moves a player to the left as far as possible (see property (ii) of the algorithm), we have that if the algorithm moves player k to a later component, then the players from coalition S that player k jumps over all have a strict higher urgency than player k .

In the following example the algorithm is applied on an instance of a SoSi sequencing game. In this example we use the concept of composed costs per time unit and composed processing times, where the *composed costs per time unit* α_U and the *composed processing time* p_U for a coalition $U \in 2^N$ are defined by

$$\alpha_U = \sum_{i \in U} \alpha_i,$$

and

$$p_U = \sum_{i \in U} p_i,$$

respectively.

Example 3.2. Consider a one-machine sequencing situation (N, σ_0, p, α) with $S \subseteq N$ such that $S = \{1, 2, \dots, 10\}$. In Figure 4 an illustration can be found of initial order σ_0 together with all relevant data on the cost coefficients and processing times (the numbers above and below the players, respectively). The completion times of the players with respect to this initial order are also indicated in the figure (bottom line in bold).

	4	3	5		6	3		3		4	2	2	7		α_i
$\overline{S}_0^{\sigma_0}$	1	2	3	$\overline{S}_1^{\sigma_0}$	4	5	$\overline{S}_2^{\sigma_0}$	6	$\overline{S}_3^{\sigma_0}$	7	8	9	10	$\overline{S}_4^{\sigma_0}$	σ_0
1	9	6	4	5	4	9	5	8	3	9	10	8	7	2	p_i
	1	10	16	20	25	29	38	43	51	54	63	73	81	88	90 $C_i(\sigma_0)$

Figure 4: Initial order σ_0 in Example 3.2

In the preprocessing step the algorithm processing order σ is set to processing order σ_0^S (see Figure 5) and we initialize

$$v(S) := \sum_{i \in S} \alpha_i (C_i(\sigma_0) - C_i(\sigma_0^S)) = 187.$$

Next, the players in S are considered in reverse order with respect σ_0^S and the algorithm starts with the last player of the penultimate component, which is player 6.

	5	3	4		6	3		3		7	4	2	2		α_i
$\overline{S}_0^{\sigma_0}$	3	2	1	$\overline{S}_1^{\sigma_0}$	4	5	$\overline{S}_2^{\sigma_0}$	6	$\overline{S}_3^{\sigma_0}$	10	7	9	8	$\overline{S}_4^{\sigma_0}$	σ_0^S
1	4	6	9	5	4	9	5	8	3	7	9	8	10	2	p_i
	1	5	11	20	25	29	38	43	51	54	61	70	78	88	90 $C_i(\sigma_0^S)$

Figure 5: The processing order σ after the preprocessing step in Example 3.2

Player 6: If player 6 is moved to the last component, then the position of player 6 should be behind player 7 (since the players in the components must stay in non-increasing order with respect to their urgencies, see property (i) of the algorithm). The resulting cost savings are

$$\alpha_{S^\sigma(6,7]} p_6 - \alpha_6 p_{N^\sigma(6,7]} = (\alpha_{10} + \alpha_7) p_6 - \alpha_6 (p_{\overline{S}_3^{\sigma_0}} + p_{10} + p_7) = 11 \cdot 8 - 3 \cdot 19 = 31.$$

Hence, we update processing order σ by moving player 6 to the position directly behind player 7 (see Figure 6) and we set $v(S) := 187 + 31 = 218$.

$\overline{S}_0^{\sigma_0}$	3	2	1	$\overline{S}_1^{\sigma_0}$	4	5	$\overline{S}_2^{\sigma_0}$	$\overline{S}_3^{\sigma_0}$	10	7	6	9	8	$\overline{S}_4^{\sigma_0}$
-----------------------------	---	---	---	-----------------------------	---	---	-----------------------------	-----------------------------	----	---	---	---	---	-----------------------------

Figure 6: The processing order σ after player 6 is considered in Example 3.2

Player 5: According to the given urgencies, player 5 should be moved to the position directly behind player 6 if he is moved to a later component. The resulting cost savings are

$$\alpha_{S^\sigma(5,6]} p_5 - \alpha_5 p_{N^\sigma(5,6]} = (\alpha_{10} + \alpha_7 + \alpha_6) p_5 - \alpha_5 (p_{\overline{S}_2^{\sigma_0}} + p_{\overline{S}_3^{\sigma_0}} + p_{10} + p_7 + p_6) = 14 \cdot 9 - 3 \cdot 32 = 30.$$

$\overline{S}_0^{\sigma_0}$	3	2	1	$\overline{S}_1^{\sigma_0}$	4	$\overline{S}_2^{\sigma_0}$	$\overline{S}_3^{\sigma_0}$	10	7	6	5	9	8	$\overline{S}_4^{\sigma_0}$
-----------------------------	---	---	---	-----------------------------	---	-----------------------------	-----------------------------	----	---	---	---	---	---	-----------------------------

Figure 7: The processing order σ after player 5 is considered in Example 3.2

Hence, we update processing order σ by moving player 5 to the position directly behind player 6 (see Figure 7) and we set $v(S) := 218 + 30 = 248$.

Player 4: Since all followers of player 4 who are members of S have a lower urgency, it is impossible to reduce the total costs by moving player 4 to a different position (see Proposition (3.1)). Hence, σ and $v(S)$ are not changed.

Player 1: There are two components behind player 1. If player 1 is moved to a different component, then the position of player 1 should be either directly behind player 4 or directly behind player 10. The resulting cost savings are 18 and 21, respectively. Hence, player 1 is moved behind player 10 (see property (ii) of the algorithm). Processing order σ is updated (see Figure 8) and $v(S)$ is increased by 21, so $v(S) := 269$.

$\overline{S}_0^{\sigma_0}$	3	2	$\overline{S}_1^{\sigma_0}$	4	$\overline{S}_2^{\sigma_0}$	$\overline{S}_3^{\sigma_0}$	10	1	7	6	5	9	8	$\overline{S}_4^{\sigma_0}$
-----------------------------	---	---	-----------------------------	---	-----------------------------	-----------------------------	----	---	---	---	---	---	---	-----------------------------

Figure 8: The processing order σ after player 1 is considered in Example 3.2

Player 2: Like in the previous step we have again two possibilities, namely moving behind player 4 with cost savings 9 or behind player 10 with cost savings 6. Hence, it is most beneficial to move player 2 behind player 4. Processing order σ is updated (see Figure 9) and $v(S)$ is increased by 9, so $v(S) := 278$.

$\overline{S}_0^{\sigma_0}$	3	$\overline{S}_1^{\sigma_0}$	4	2	$\overline{S}_2^{\sigma_0}$	$\overline{S}_3^{\sigma_0}$	10	1	7	6	5	9	8	$\overline{S}_4^{\sigma_0}$
-----------------------------	---	-----------------------------	---	---	-----------------------------	-----------------------------	----	---	---	---	---	---	---	-----------------------------

Figure 9: The processing order σ after player 2 is considered in Example 3.2

Player 3: There are two components behind player 3. Note that all players in the last component have a lower urgency than player 3. Therefore, it is impossible to reduce the total costs by moving player 3 to the last component. If player 3 is moved to the second component, then the position of player 3 should be directly behind player 4. The resulting cost savings are -21 and thus moving player 3 to the second component will not reduce the total costs. Hence, the order depicted in Figure 9 is the optimal processing order σ_S for coalition S obtained by the algorithm. Furthermore, $v(S) = 278$. \triangle

The following proposition, which will frequently be used later on, provides a basic property of composed costs per time unit and composed processing times. Namely, if every player in a set of players U are individually more urgent than a specific player i , then also the composed job U as a whole is more urgent than player i .

Proposition 3.2. *Let $U \subsetneq N$ with $U \neq \emptyset$ and let $i \in N \setminus U$. If $u_i < u_j$ for all $j \in U$, then*

$$\frac{\alpha_i}{p_i} < \frac{\alpha_U}{p_U},$$

or equivalently,

$$\alpha_i p_U - \alpha_U p_i < 0.^2$$

Proof. Assume $u_i < u_j$ for all $j \in U$, i.e., $\alpha_i p_j < \alpha_j p_i$, for all $j \in U$. By adding these $|U|$ equations we get $\alpha_i \sum_{j \in U} p_j < p_i \sum_{j \in U} \alpha_j$, i.e.,

$$\frac{\alpha_i}{p_i} < \frac{\sum_{j \in U} \alpha_j}{\sum_{j \in U} p_j} = \frac{\alpha_U}{p_U}. \quad \square$$

The following lemma compares the processing orders that are obtained from the algorithm with respect to coalition S and coalition $S \cup \{i\}$, in case player $i \in N \setminus S$ is the only player in the component of $S \cup \{i\}$ with respect to σ_0 . This lemma will be the driving force behind Theorem 3.4, which in turn is the crux for proving convexity of SoSi sequencing games.

Lemma 3.3. *Let (N, σ_0, p, α) be a one-machine sequencing situation, let $S \subsetneq N$ with $S \neq \emptyset$ and let $i \in N \setminus S$ be such that $(S \cup \{i\})_{c(i, S \cup \{i\}, \sigma_0)}^{\sigma_0} = \{i\}$. Then, for all $k \in S$ we have*

$$c(k, S \cup \{i\}, \sigma_{S \cup \{i\}}) \geq c(k, S \cup \{i\}, \sigma_S).$$

Proof. See Appendix A. \square

From the previous lemma it follows that if one wants to determine an optimal processing order of a coalition in a SoSi sequencing game, then the information of optimal processing orders of specific subcoalitions can be used. More precisely, if one wants to know the optimal processing order $\sigma_{S \cup \{i\}}$ derived by the algorithm for a coalition $S \cup \{i\}$ with $i \notin S$ and i being the only player in its component in σ_0 , then it does not matter whether you take σ_0 or σ_S as initial processing order, as is stated in the following theorem.

Since the initial order will be varied we need some additional notation. We denote the obtained processing order after the complete run of the algorithm for one-machine sequencing situation (N, σ, p, α) with initial order σ and coalition S by $\text{Alg}((N, \sigma, p, \alpha), S)$. Hence, $\text{Alg}((N, \sigma_0, p, \alpha), S) = \sigma_S$.

Theorem 3.4. *Let (N, σ_0, p, α) be a one-machine sequencing situation, let $S \subsetneq N$ with $S \neq \emptyset$ and let $i \in N \setminus S$ be such that $(S \cup \{i\})_{c(i, S \cup \{i\}, \sigma_0)}^{\sigma_0} = \{i\}$. Then,*

$$\sigma_{S \cup \{i\}} = \text{Alg}((N, \sigma_S, p, \alpha), S \cup \{i\}).$$

Proof. We start with proving that the minimum costs for coalition $S \cup \{i\}$ in the sequencing situation (N, σ_0, p, α) is equal to the minimum costs for coalition $S \cup \{i\}$ in the sequencing situation (N, σ_S, p, α) . Then, we show that the two corresponding sets of optimal processing orders are equal. Finally, the fact that the algorithm always selects a unique processing order among the set of all optimal processing orders (property (iv)) completes the proof.

Note that $\mathcal{A}(\sigma_S, S \cup \{i\}) \subseteq \mathcal{A}(\sigma_0, S \cup \{i\})$ and thus

$$\min_{\sigma \in \mathcal{A}(\sigma_0, S \cup \{i\})} \sum_{j \in S \cup \{i\}} \alpha_j C_j(\sigma) \leq \min_{\sigma \in \mathcal{A}(\sigma_S, S \cup \{i\})} \sum_{j \in S \cup \{i\}} \alpha_j C_j(\sigma). \quad (2)$$

²Note that this proposition also holds if every $<$ sign is replaced by a $>$, \leq or \geq sign.

Moreover, from Lemma 3.3 we know that for all $k \in S \cup \{i\}$ we have $c(k, S \cup \{i\}, \sigma_{S \cup \{i\}}) \geq c(k, S \cup \{i\}, \sigma_S)$ and thus $\sigma_{S \cup \{i\}} \in \mathcal{A}(\sigma_S, S \cup \{i\})$. As a consequence, since

$$\sum_{j \in S \cup \{i\}} \alpha_j C_j(\sigma_{S \cup \{i\}}) = \min_{\sigma \in \mathcal{A}(\sigma_0, S \cup \{i\})} \sum_{j \in S \cup \{i\}} \alpha_j C_j(\sigma),$$

we have together with (2) that

$$\min_{\sigma \in \mathcal{A}(\sigma_0, S \cup \{i\})} \sum_{j \in S \cup \{i\}} \alpha_j C_j(\sigma) = \min_{\sigma \in \mathcal{A}(\sigma_S, S \cup \{i\})} \sum_{j \in S \cup \{i\}} \alpha_j C_j(\sigma). \quad (3)$$

Let $\mathcal{O}(\sigma_0, S \cup \{i\})$ and $\mathcal{O}(\sigma_S, S \cup \{i\})$ denote the set of optimal processing orders for coalition $S \cup \{i\}$ in sequencing situation (N, σ_0, p, α) and (N, σ_S, p, α) respectively. We will show $\mathcal{O}(\sigma_0, S \cup \{i\}) = \mathcal{O}(\sigma_S, S \cup \{i\})$.

First, take $\sigma^* \in \mathcal{O}(\sigma_S, S \cup \{i\})$. Since $\mathcal{A}(\sigma_S, S \cup \{i\}) \subseteq \mathcal{A}(\sigma_0, S \cup \{i\})$, we have $\sigma^* \in \mathcal{A}(\sigma_0, S \cup \{i\})$. Moreover, due to (3), we also have $\sigma^* \in \mathcal{O}(\sigma_0, S \cup \{i\})$.

Second, take $\sigma^* \in \mathcal{O}(\sigma_0, S \cup \{i\})$. From property (iv) of the algorithm we know that for all $k \in S \cup \{i\}$ we have $c(k, S \cup \{i\}, \sigma^*) \geq c(k, S \cup \{i\}, \sigma_{S \cup \{i\}})$. Therefore, together with $c(k, S \cup \{i\}, \sigma_{S \cup \{i\}}) \geq c(k, S \cup \{i\}, \sigma_S)$ from Lemma 3.3, we know $\sigma^* \in \mathcal{A}(\sigma_S, S \cup \{i\})$. Consequently, together with (3), we can conclude $\sigma^* \in \mathcal{O}(\sigma_S, S \cup \{i\})$. Hence, we have

$$\mathcal{O}(\sigma_0, S \cup \{i\}) = \mathcal{O}(\sigma_S, S \cup \{i\}).$$

Finally, since the algorithm chooses among all optimal processing orders the order in which the players are in a component to the left as far as possible and because the algorithm chooses a fixed order within the components (property (iv)), we have $\sigma_{S \cup \{i\}} = \text{Alg}((N, \sigma_S, p, \alpha), S \cup \{i\})$. \square

It readily follows from the previous theorem that all players in a component to the right of player i with respect to σ_S are not moved to a different component when applying the algorithm to one-machine sequencing situation (N, σ_S, p, α) and coalition $S \cup \{i\}$. This is stated in the following proposition.

Proposition 3.5. *Let (N, σ_0, p, α) be a one-machine sequencing situation, let $S \subsetneq N$ with $S \neq \emptyset$ and let $i \in N \setminus S$ be such that $(S \cup \{i\})_{c(i, S \cup \{i\}, \sigma_0)}^{\sigma_0} = \{i\}$. Then for all $k \in S \cap F(\sigma_S, i)$ we have*

$$c(k, S \cup \{i\}, \sigma_{S \cup \{i\}}) = c(k, S \cup \{i\}, \sigma_S),$$

The next proposition states that all players in a component to the left of player i with respect to σ_S are, if they are moved by the algorithm, moved componentwise at least as far as the original component of player i in σ_0 . As a consequence, all players that are in $\sigma_{S \cup \{i\}}$ to the left of the original component of player i in σ_0 , are not moved by the algorithm when going from σ_S to $\sigma_{S \cup \{i\}}$.

Proposition 3.6. *Let (N, σ_0, p, α) be a one-machine sequencing situation, let $S \subsetneq N$ with $S \neq \emptyset$ and let $i \in N \setminus S$ be such that $(S \cup \{i\})_{c(i, S \cup \{i\}, \sigma_0)}^{\sigma_0} = \{i\}$.*

(i) *For all $k \in S$ with $c(k, S \cup \{i\}, \sigma_{S \cup \{i\}}) > c(k, S \cup \{i\}, \sigma_S)$ we have*

$$c(k, S \cup \{i\}, \sigma_{S \cup \{i\}}) \geq c(i, S \cup \{i\}, \sigma_0),$$

(ii) For all $k \in S$ with $c(k, S \cup \{i\}, \sigma_{S \cup \{i\}}) < c(i, S \cup \{i\}, \sigma_0)$ we have

$$c(k, S \cup \{i\}, \sigma_{S \cup \{i\}}) = c(k, S \cup \{i\}, \sigma_S).$$

The previous proposition follows directly from the following, more technical, lemma. This lemma shows that, when applying the algorithm to one-machine sequencing situation (N, σ_S, p, α) and coalition $S \cup \{i\}$, once a predecessor of player i with respect to σ_S is considered by the algorithm, moving this player to a position that is to the left of the original component of player i in σ_0 is never beneficial.

Lemma 3.7. *Let (N, σ_0, p, α) be a one-machine sequencing situation, let $S \subsetneq N$ with $S \neq \emptyset$ and let $i \in N \setminus S$ be such that $(S \cup \{i\})_{c(i, S \cup \{i\}, \sigma_0)}^{\sigma_0} = \{i\}$. Let $m \in S \cap P(\sigma_S, i)$ and $l \in S \cap F(\tau_m, m)$ with $c(l, S \cup \{i\}, \tau_m) < c(i, S \cup \{i\}, \sigma_0)$. Then*

$$\alpha_{(S \cup \{i\})^{\tau_m}(m, l]} p_m - \alpha_m p_{N^{\tau_m}(m, l]} \leq 0, \quad (4)$$

where τ_m denotes the processing order during the run of the algorithm for one-machine sequencing situation (N, σ_S, p, α) and coalition $S \cup \{i\}$ just before player m is considered.

Proof. See Appendix B. □

4 On the convexity of SoSi sequencing games

A game $v \in \text{TU}^N$ is called convex if

$$v(S \cup \{i\}) - v(S) \leq v(T \cup \{i\}) - v(T), \quad (5)$$

for all $S, T \in 2^N \setminus \{\emptyset\}$, $i \in N$ such that $S \subset T \subseteq N \setminus \{i\}$, i.e., the incentive for joining a coalition increases as the coalition grows. Using recursive arguments it can be seen that in order to prove convexity it is sufficient to show 5 for the case $|T| = |S| + 1$ which boils down to

$$v(S \cup \{i\}) - v(S) \leq v(S \cup \{j\} \cup \{i\}) - v(S \cup \{j\}), \quad (6)$$

for all $S \in 2^N \setminus \{\emptyset\}$, $i, j \in N$ and $i \neq j$ such that $S \subseteq N \setminus \{i, j\}$.

The main result of this paper is the following theorem.

Theorem 4.1. *Let (N, σ_0, p, α) be a one-machine sequencing situation and let (N, v) be the corresponding SoSi sequencing game. Then, (N, v) is convex.*

Before presenting the formal proof of our main result, we highlight some of its important aspects beforehand. Using (6), let $S \in 2^N \setminus \{\emptyset\}$, $i, j \in N$ and let $i \neq j$ be such that $S \subseteq N \setminus \{i, j\}$.

Note that without loss of generality we can assume

- Assumption 1: $\sigma_0(j) < \sigma_0(i)$.
- Assumption 2: $(S \cup \{j\} \cup \{i\})_{c(j, S \cup \{j\} \cup \{i\}, \sigma_0)}^{\sigma_0} = \{j\}$ and $(S \cup \{j\} \cup \{i\})_{c(i, S \cup \{j\} \cup \{i\}, \sigma_0)}^{\sigma_0} = \{i\}$.

The first assumption is harmless because of the symmetric role of i and j in (6). The second assumption states that player i and j both are the only player in the component of $S \cup \{j\} \cup \{i\}$ with respect to σ_0 . In theory this is no restriction since it is always possible to add dummy players with zero processing times and zero costs per time unit

(a more formal explanation can be found in Appendix C). This assumption facilitates the comparison of the marginal contribution of player i to coalition S and the marginal contribution of player i to coalition $S \cup \{j\}$. For example, if one determines the optimal processing order for coalition $S \cup \{i\}$ via initial processing order σ_S and player i is the only player in its component of $S \cup \{j\} \cup \{i\}$ with respect to σ_0 (and thus also with respect to σ_S), then the players of coalition $S \cup \{i\}$ are in every component already ordered with respect to their urgency and thus the preprocessing step of the algorithm can be skipped. As a consequence, the marginal contribution of player i to coalition S can be written as the sum of the positive cost difference of the players who are moved by the algorithm to a different component.

In order to denote the different types of players that are moved, we introduce the following notation. For $U \in 2^N \setminus \{\emptyset\}$ and $k \in N$ such that $U \subseteq N \setminus \{k\}$ and $(U \cup \{k\})_{c(k, U \cup \{k\}, \sigma_0)}^{\sigma_0} = \{k\}$, let $M^k(U)$ denote the set of players who are moved to a different component during the run of the algorithm with respect to one-machine sequencing situation (N, σ_U, p, α) and coalition $U \cup \{k\}$. Since the algorithm only moves players to components that are to the right of its original component in σ_U , we have

$$M^k(U) = \{l \in N \mid c(l, \sigma_{U \cup \{k\}}) > c(l, \sigma_U)\}.$$

As the algorithm only moves the players of the coalition $U \cup \{k\}$ and all players outside this coalition are not moved, we have

$$M^k(U) \subseteq (U \cup \{k\}). \quad (7)$$

Moreover, from Proposition 3.5 and 3.6 it follows respectively that

$$M^k(U) \subseteq (P(\sigma_U, k) \cup \{k\}), \quad (8)$$

and

$$c(l, \sigma_{U \cup \{k\}}) \geq c(k, \sigma_U), \quad (9)$$

for all $l \in M^k(U)$.

In order to prove Theorem 4.1 we need to compare the marginal contribution of player i to coalition S and the marginal contribution of player i to coalition $S \cup \{j\}$. As argued above, both marginal contributions can be written as the sum of the positive cost differences of the players who are moved by the algorithm to a different component. In order to compare those cost differences more easily, we first partition the players in $M^i(S)$, based on their position in the processing orders σ_S and $\sigma_{S \cup \{j\}}$, in four subsets. Second, we derive from σ_S a special processing order $\bar{\sigma}$ in such a way that all players from $M^i(S)$ are in $\bar{\sigma}$ and $\sigma_{S \cup \{j\}}$ in the same component. The convexity proof is finished by means of adequately comparing all positive cost differences.

Proof of Theorem 4.1. Let $S \in 2^N \setminus \{\emptyset\}$, $i, j \in N$ and $i \neq j$ such that $S \subseteq N \setminus \{i, j\}$. We will prove

$$v(S \cup \{i\}) - v(S) \leq v(S \cup \{j\} \cup \{i\}) - v(S \cup \{j\}). \quad (10)$$

We partition the players in $M^i(S)$, based on their position in the processing orders σ_S and $\sigma_{S \cup \{j\}}$, in four subsets. First, note that from (7) it follows that $M^i(S) \subseteq S \cup \{i\}$ and thus $j \notin M^i(S)$. From (8) it follows that all players in $M^i(S)$ are in σ_S to the left of player i , or player i himself. By assumption 1 we have that player j is to the left of player i in σ_0 (and thus also in σ_S). So, we can split $M^i(S)$ into the following two disjoint sets:

- $M_1^i(S)$: the set of players in $M^i(S)$ who are in σ_S to the left of player j ,
- $M_2^i(S)$: the set of players in $M^i(S)$ who are in σ_S between player j and player i , or player i himself.

Based on the position in $\sigma_{S \cup \{j\}}$, we can split $M_1^i(S)$ into another three disjoint subsets:

- $M_{1a}^i(S)$: the set of players in $M_1^i(S)$ who are in $\sigma_{S \cup \{j\}}$ to the left of the original component of player j ,
- $M_{1b}^i(S)$: the set of players in $M_1^i(S)$ who are in $\sigma_{S \cup \{j\}}$ between the original components of player j and player i , or in the original component of player j ,
- $M_{1c}^i(S)$: the set of players in $M_1^i(S)$ who are in $\sigma_{S \cup \{j\}}$ to the right of the original component of player i .

From Proposition 3.5 it follows that all players in $M_2^i(S)$ are in $\sigma_{S \cup \{j\}}$ between the original components of player j and player i , so we do not further split $M_2^i(S)$ into subsets. We have now a partition of $M^i(S)$ in four subsets, namely $\{M_{1a}^i(S), M_{1b}^i(S), M_{1c}^i(S), M_2^i(S)\}$. Moreover, if $i \in M^i(S)$ then $i \in M_2^i(S)$.

The definition of the partition of $M^i(S)$ in four subsets explains the position of the corresponding players in the processing orders σ_S and $\sigma_{S \cup \{j\}}$. The following four claims indicate how the partition also determines the position in the two other processing orders $\sigma_{S \cup \{i\}}$ and $\sigma_{S \cup \{j\} \cup \{i\}}$. For notational convenience, we denote $c(k, S \cup \{i\} \cup \{j\}, \sigma)$ by $c(k, \sigma)$ for every $k \in S \cup \{i\} \cup \{j\}$ and $\sigma \in \Pi(N)$.

- **Claim 1:** $c(k, \sigma_{S \cup \{j\} \cup \{i\}}) = c(k, \sigma_{S \cup \{i\}}) \geq c(i, \sigma_S)$ for all $k \in M^i(S)$.
- **Claim 2:** $c(k, \sigma_{S \cup \{j\} \cup \{i\}}) = c(k, \sigma_{S \cup \{j\}})$ for all $k \in M_{1c}^i(S)$.
- **Claim 3:** $c(k, \sigma_S) = c(k, \sigma_{S \cup \{j\}})$ for all $k \in M_2^i(S)$.
- **Claim 4:** $c(k, \sigma_S) = c(k, \sigma_{S \cup \{j\}})$ for all $k \in M_{1a}^i(S)$.

The proofs of these four claims can be found in Appendix D. Figure 10 illustrates for all four partition elements of $M^i(S)$ its position with respect to the original components of player i and player j in the four different processing orders. The solid arrows give the original components and/or the actual positions of player i and j . The dotted arrows give possible positions of player i or j .

We define $\bar{\sigma} \in \Pi(N)$ as the unique urgency respecting processing order that satisfies

- (i) for all $k \in M^i(S)$:

$$c(k, \bar{\sigma}) = c(k, \sigma_{S \cup \{j\}}), \quad (11)$$

- (ii) for all $k \in S \setminus M^i(S)$:

$$c(k, \bar{\sigma}) = c(k, \sigma_S),$$

- (iii) for all $k, l \in S$ with $c(k, \bar{\sigma}) = c(l, \bar{\sigma})$:

$$u_k = u_l, \sigma_0(k) < \sigma_0(l) \Rightarrow \bar{\sigma}(k) < \bar{\sigma}(l).$$

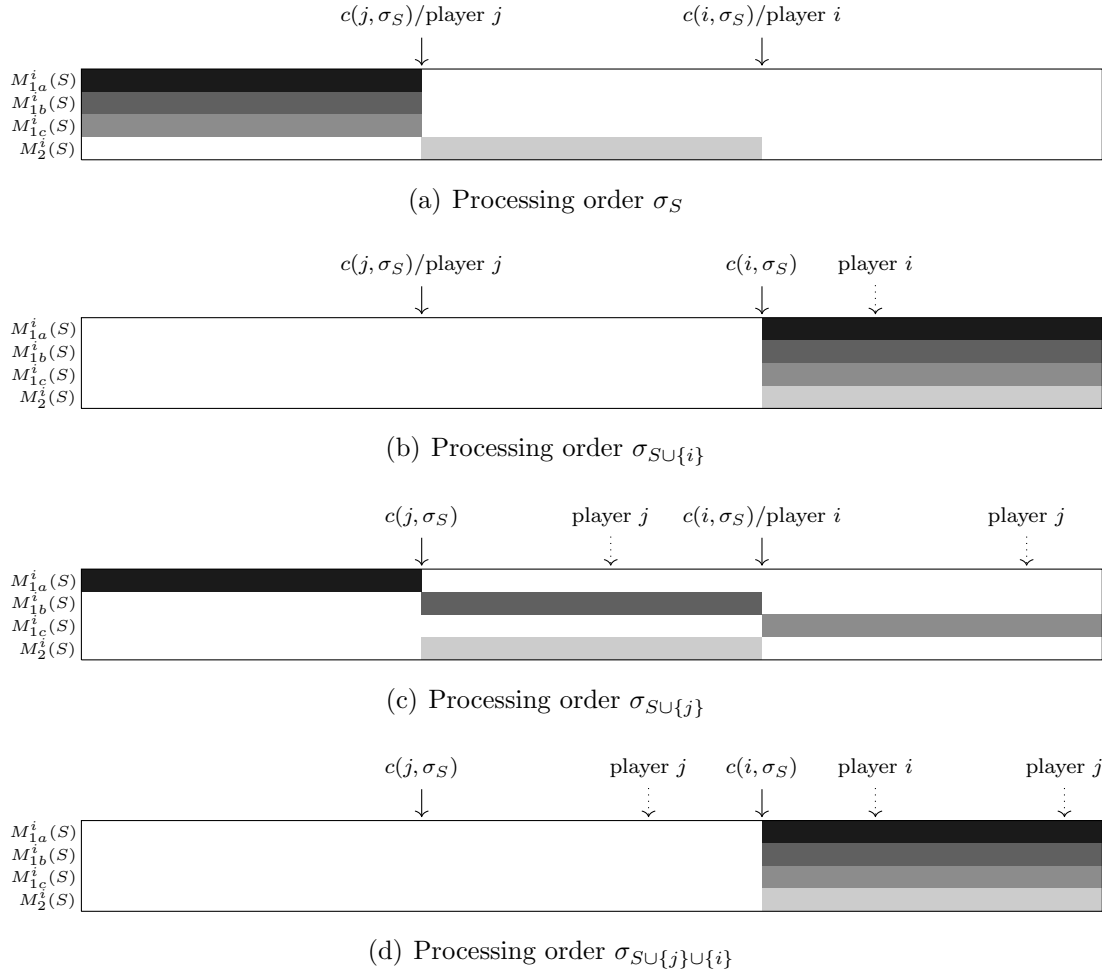


Figure 10: The position of the players in $M^i(S)$ in the four different processing orders

Note that conditions (i) and (ii) determine the components for the players in S . Next, the urgency respecting requirement determines the order within the components for the players with different urgencies. Finally, in case there is a tie for the urgency of two players in the same component, item (iii) states a tiebreaking rule. As a consequence, due to this tiebreaking rule, we have that $\bar{\sigma}$ is unique.

Note that $\bar{\sigma}$ can be considered as a temporary processing order when going from σ_S to $\sigma_{S \cup \{i\}}$ (cf. Figure 11). The processing order $\bar{\sigma}$ is derived from processing order σ_S in such a way that all players from $M^i(S)$ are in $\bar{\sigma}$ and $\sigma_{S \cup \{j\}}$ in the same component. From claim 3 and 4 it follows that the players in $M^i_{1a}(S)$ and $M^i_2(S)$ are in σ_S and $\sigma_{S \cup \{j\}}$ in the same component and thus those players do not need to be moved. Hence, only the players in $M^i_{1b}(S)$ and $M^i_{1c}(S)$ need to be moved. Hence, we start from σ_S and we move all players in $M^i_{1b}(S)$ and $M^i_{1c}(S)$ to the components they are in in $\sigma_{S \cup \{j\}}$. Note that since the tiebreaking rule mentioned in condition (iii) is the same tiebreaking rule as in property (iv) of the algorithm, the mutual order of the players in $M^i(S)$ is in $\bar{\sigma}$ the same as in $\sigma_{S \cup \{j\}}$.

An illustration of the position of the players in $M^i(S)$ in $\bar{\sigma}$ can be found in Figure 12. Note that since $i \notin S \cup \{j\}$ it follows that $c(i, \sigma_S) = c(i, \sigma_{S \cup \{j\}})$. Moreover, we note that $\bar{\sigma}$ and $\sigma_{S \cup \{j\}}$ are not necessarily equal to each other as the players in $M^j(S) \setminus M^i(S)$ are in $\bar{\sigma}$ and $\sigma_{S \cup \{j\}}$ in different components. However, as the players in $M^j(S)$ will be moved

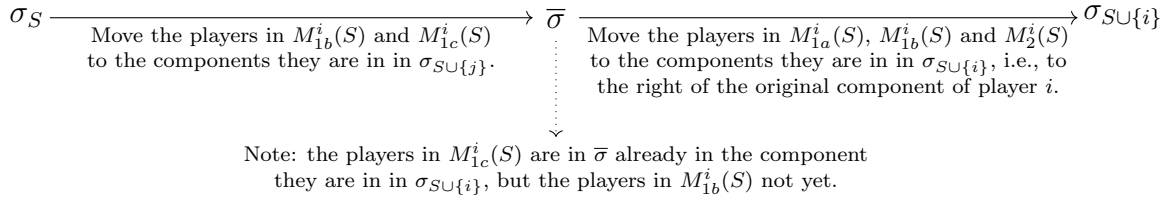


Figure 11: Overview how to obtain $\sigma_{S \cup \{i\}}$ from σ_S via $\bar{\sigma}$

to a component to the right when going from σ_S to $\sigma_{S \cup \{j\}}$, we have

$$c(k, \sigma_{S \cup \{j\}}) \geq c(k, \bar{\sigma}), \quad (12)$$

for all $k \in S \cup \{j\} \cup \{i\}$.

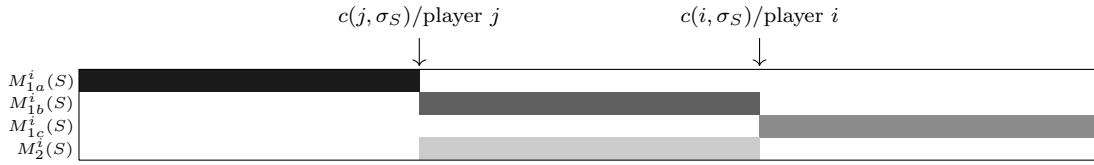


Figure 12: The position of the players in $M^i(S)$ in $\bar{\sigma}$

Now we consider the transition from $\bar{\sigma}$ to $\sigma_{S \cup \{i\}}$ and its corresponding cost differences. Since the players in $M_{1c}^i(S)$ are in $\bar{\sigma}$ already in the component they are in in $\sigma_{S \cup \{i\}}$, only all players in $M_{1a}^i(S)$, $M_{1b}^i(S)$ and $M_2^i(S)$ need to be moved to a component to the right when going from $\bar{\sigma}$ to $\sigma_{S \cup \{i\}}$ (see also Figure 11). We go from $\bar{\sigma}$ to $\sigma_{S \cup \{i\}}$ by considering the players in $M_{1a}^i(S)$, $M_{1b}^i(S)$ and $M_2^i(S)$ in an order reverse to the order they are in $\bar{\sigma}$, i.e., the players are considered from the right to the left. For $k \in M_{1a}^i(S) \cup M_{1b}^i(S) \cup M_2^i(S)$, denote the processing order just before player k is moved by $\bar{\tau}_k$ and let \bar{r}_k denote the player where player k will be moved behind. The cost difference for coalition $S \cup \{i\}$ due to moving this player, when going from $\bar{\sigma}$ to $\sigma_{S \cup \{i\}}$, is denoted by $\bar{\delta}_k$, i.e.,

$$\bar{\delta}_k = \alpha_{(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{r}_k)} p_k - \alpha_k p_{N^{\bar{\tau}_k}(k, \bar{r}_k)}.$$

Similarly, we can write the marginal contribution of player i to coalition $S \cup \{j\}$ as the sum of positive cost differences of the players in $M^i(S \cup \{j\})$. We go from $\sigma_{S \cup \{j\}}$ to $\sigma_{S \cup \{j\} \cup \{i\}}$ by considering the players in $M^i(S \cup \{j\})$ in an order reverse to the order they are in $\sigma_{S \cup \{j\}}$, i.e., the players are considered from the right to the left. We note that since the mutual order of the players in $M^i(S)$ is the same in $\bar{\sigma}$ and $\sigma_{S \cup \{j\}}$, the order in which the players in $M_{1a}^i(S) \cup M_{1b}^i(S) \cup M_2^i(S)$ are considered when going from $\bar{\sigma}$ to $\sigma_{S \cup \{i\}}$ is the same as the order in which they are considered when going from $\sigma_{S \cup \{j\}}$ to $\sigma_{S \cup \{j\} \cup \{i\}}$. For $k \in M^i(S \cup \{j\})$, denote the processing order just before player k is moved by τ_k and let r_k denote the player where player k will be moved behind. The cost difference for coalition $S \cup \{j\} \cup \{i\}$ due to moving this player, when going from $\sigma_{S \cup \{j\}}$ to $\sigma_{S \cup \{j\} \cup \{i\}}$, is denoted by δ_k , i.e.,

$$\delta_k = \alpha_{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k)} p_k - \alpha_k p_{N^{\tau_k}(k, r_k)}.$$

From (12) together with the fact that the players in $M_{1a}^i(S) \cup M_{1b}^i(S) \cup M_2^i(S)$ are moved to the same component in $\sigma_{S \cup \{i\}}$ and $\sigma_{S \cup \{j\} \cup \{i\}}$, and the fact that the players in

$M^j(S) \setminus M^i(S)$ are moved to a component to the right, we have

$$c(l, \tau_k) \geq c(l, \bar{\tau}_k), \quad (13)$$

for all $k \in M_{1a}^i(S) \cup M_{1b}^i(S) \cup M_2^i(S)$ and $l \in S \cup \{j\} \cup \{i\}$.

The following claim states that the cost savings when moving a player in $M_{1a}^i(S) \cup M_{1b}^i(S) \cup M_2^i(S)$ when going from $\bar{\sigma}$ to $\sigma_{S \cup \{i\}}$ is at most the cost savings when moving the same player when going from $\sigma_{S \cup \{j\}}$ to $\sigma_{S \cup \{j\} \cup \{i\}}$.

Claim 5: $\bar{\delta}_k \leq \delta_k$ for all $k \in M_{1a}^i(S) \cup M_{1b}^i(S) \cup M_2^i(S)$.

Proof. The proof can be found in Appendix E.

We are now ready to prove (10). Note that a detailed explanation of the subsequent equalities and inequalities can be found after the equations.

$$\begin{aligned}
& v(S \cup \{i\}) - v(S) \\
& \stackrel{(i)}{=} \sum_{k \in P(\sigma_0, i) \cap F(\sigma_S, i)} \alpha_i p_k + C(\sigma_S, S \cup \{i\}) - C(\sigma_{S \cup \{i\}}, S \cup \{i\}) \\
& \stackrel{(ii)}{=} \sum_{k \in P(\sigma_0, i) \cap F(\sigma_S, i)} \alpha_i p_k + C(\sigma_S, S \cup \{i\}) - C(\bar{\sigma}, S \cup \{i\}) \\
& \quad + C(\bar{\sigma}, S \cup \{i\}) - C(\sigma_{S \cup \{i\}}, S \cup \{i\}) \\
& \stackrel{(iii)}{=} \sum_{k \in P(\sigma_0, i) \cap F(\sigma_S, i)} \alpha_i p_k + C(\sigma_S, S) - C(\bar{\sigma}, S) + \sum_{k \in M_{1c}^i(S)} \alpha_i p_k \\
& \quad + C(\bar{\sigma}, S \cup \{i\}) - C(\sigma_{S \cup \{i\}}, S \cup \{i\}) \\
& \stackrel{(iv)}{\leq} \sum_{k \in P(\sigma_0, i) \cap F(\sigma_S, i)} \alpha_i p_k + \sum_{k \in M_{1c}^i(S)} \alpha_i p_k + C(\bar{\sigma}, S \cup \{i\}) - C(\sigma_{S \cup \{i\}}, S \cup \{i\}) \\
& \stackrel{(v)}{=} \sum_{k \in P(\sigma_0, i) \cap F(\sigma_S, i)} \alpha_i p_k + \sum_{k \in M_{1c}^i(S)} \alpha_i p_k + \sum_{k \in M_{1a}^i(S) \cup M_{1b}^i(S) \cup M_2^i(S)} \bar{\delta}_k \\
& \stackrel{(vi)}{\leq} \sum_{k \in P(\sigma_0, i) \cap F(\sigma_S, i)} \alpha_i p_k + \sum_{k \in M_{1c}^i(S)} \alpha_i p_k + \sum_{k \in M_{1a}^i(S) \cup M_{1b}^i(S) \cup M_2^i(S)} \delta_k \\
& \stackrel{(vii)}{\leq} \sum_{k \in P(\sigma_0, i) \cap F(\sigma_S, i)} \alpha_i p_k + \sum_{k \in M_{1c}^i(S)} \alpha_i p_k + \sum_{k \in M^i(S \cup \{j\})} \delta_k \\
& \stackrel{(viii)}{=} \sum_{k \in P(\sigma_0, i) \cap F(\sigma_S, i)} \alpha_i p_k + \sum_{k \in M_{1c}^i(S)} \alpha_i p_k \\
& \quad + C(\sigma_{S \cup \{j\}}, S \cup \{j\} \cup \{i\}) - C(\sigma_{S \cup \{j\} \cup \{i\}}, S \cup \{j\} \cup \{i\}) \\
& \stackrel{(ix)}{\leq} \sum_{k \in P(\sigma_0, i) \cap F(\sigma_S, i)} \alpha_i p_k + \sum_{k \in P(\sigma_S, i) \cap F(\sigma_{S \cup \{j\}}, i)} \alpha_i p_k \\
& \quad + C(\sigma_{S \cup \{j\}}, S \cup \{j\} \cup \{i\}) - C(\sigma_{S \cup \{j\} \cup \{i\}}, S \cup \{j\} \cup \{i\}) \\
& \stackrel{(x)}{=} \sum_{k \in P(\sigma_0, i) \cap F(\sigma_{S \cup \{j\}}, i)} \alpha_i p_k + C(\sigma_{S \cup \{j\}}, S \cup \{j\} \cup \{i\}) - C(\sigma_{S \cup \{j\} \cup \{i\}}, S \cup \{j\} \cup \{i\}) \\
& \stackrel{(xi)}{=} v(S \cup \{j\} \cup \{i\}) - v(S \cup \{j\}),
\end{aligned}$$

which proves (10).

Explanations:

- (i) The extra worth that is obtained by adding player i to coalition S can be split into two parts. The first part is due to the fact that player i joins the coalition and it represents the cost savings for player i in processing order σ_S compared to σ_0 . The completion time of player i is reduced by the sum of the processing times of the players that jumped over player i when going from σ_0 to σ_S without moving any players. The second part represents the cost savings for coalition $S \cup \{i\}$ by additionally moving players when going from σ_S to the optimal processing order $\sigma_{S \cup \{i\}}$.
- (ii) The optimal processing order $\sigma_{S \cup \{i\}}$ can be obtained from σ_S via $\bar{\sigma}$ where some players are already (partially) moved to the right.
- (iii) The cost difference for coalition $S \cup \{i\}$ when going from σ_S to $\bar{\sigma}$ can be split into two parts: the cost difference for coalition S and the cost difference for player i . By the definition of $\bar{\sigma}$ and since $i \notin S \cup \{j\}$, player i is not moved when going from σ_S to $\bar{\sigma}$ and the completion time of player i is reduced by the sum of the processing times of the players that jumped over player i when going from σ_S to $\bar{\sigma}$, i.e., the sum of the processing times of the players in $M_{1c}^i(S)$.
- (iv) Processing order σ_S is optimal for coalition S and thus $C(\sigma_S, S) - C(\bar{\sigma}, S) \leq 0$.
- (v) This follows from the definition of $\bar{\delta}_k$.
- (vi) This follows from claim 5.
- (vii) This follows from $(M_{1a}^i(S) \cup M_{1b}^i(S) \cup M_2^i(S)) \subseteq M^i(S \cup \{j\})$ (cf. Figure 10) and $\delta_k > 0$ for all $k \in M^i(S \cup \{j\})$ due to property (ii) of the algorithm.
- (viii) This follows from the definition of δ_k .
- (ix) This follows from $M_{1c}^i(S) \subseteq (P(\sigma_S, i) \cap F(\sigma_{S \cup \{j\}}, i))$ (cf. Figure 10).
- (x) The group of players that jump over player i when going from σ_0 to $\sigma_{S \cup \{j\}}$ can be split into two groups: the group of players that jumped over player i when going from σ_0 to σ_S and the group of players that were positioned in front of player i in σ_S but jumped over player i when going from σ_S to $\sigma_{S \cup \{j\}}$. Hence, $\{P(\sigma_0, i) \cap F(\sigma_S, i), P(\sigma_S, i) \cap F(\sigma_{S \cup \{j\}}, i)\}$ is a partition of $P(\sigma_0, i) \cap F(\sigma_{S \cup \{j\}}, i)$.
- (xi) Similar to the explanation in (i).

To conclude, we have shown $v(S \cup \{i\}) - v(S) \leq v(S \cup \{j\} \cup \{i\}) - v(S \cup \{j\})$ which proves the convexity of SoSi sequencing games. \square

References

- Curiel, I., G. Pederzoli, and S. Tijs (1989). Sequencing games. *European Journal of Operational Research*, **40**, 344–351.
- Curiel, I., J. Potters, R. Prasad, S. Tijs, and B. Veltman (1993). Cooperation in one machine scheduling. *Zeitschrift für Operations Research*, **38**, 113–129.
- Musegaas, M., P. Borm, and M. Quant (2015). Step out - Step in sequencing games. *European Journal of Operational Research*, **246**, 894–906.
- Slikker, M. (2006). Relaxed sequencing games have a nonempty core. *Naval Research Logistics*, **53**, 235–242.
- Smith, W. (1956). Various optimizers of single-stage production. *Naval Research Logistics Quarterly*, **3**, 59–66.
- Velzen, B. van and H. Hamers (2003). On the balancedness of relaxed sequencing games. *Mathematical Methods of Operations Research*, **57**, 287–297.

A Proof of Lemma 3.3

In this proof we denote $c(k, S \cup \{i\}, \sigma)$ by $c(k, \sigma)$ for every $k \in S \cup \{i\}$ and every $\sigma \in \Pi(N)$. We prove the lemma with the help of the algorithm. First, note that because player i is the only player in its component in σ_0 , we have $\sigma_0^S = \sigma_0^{S \cup \{i\}}$, i.e., the processing orders are the same after the preprocessing step of the algorithm. Therefore, if we go from σ_0 to the optimal processing orders σ_S and $\sigma_{S \cup \{i\}}$, then the steps performed by the algorithm are the same up to the moment that player i is considered. Moreover, since the players are considered in reverse order with respect to σ_0^S , we have

$$c(k, \sigma_{S \cup \{i\}}) = c(k, \sigma_S), \quad (14)$$

for all $k \in S \cap F(\sigma_0^S, i)$. Hence, it remains to be proven that also for the players in $S \cap P(\sigma_0^S, i)$ the lemma is true.

Let player $m \in S$ be the closest predecessor of player i with respect to σ_0^S for which the lemma is not true, i.e.,

$$c(m, \sigma_{S \cup \{i\}}) < c(m, \sigma_S),$$

and

$$c(k, \sigma_{S \cup \{i\}}) \geq c(k, \sigma_S),$$

for all $k \in S \cap F(\sigma_0^S, m) \cap P(\sigma_0^S, i)$. We will derive a contradiction. We continue the proof as follows. We look to which component player m will be moved by the algorithm with respect to coalition S . Then, there is a specific player who is in a component with index at least as high as the component that player m is moved by the algorithm with respect to coalition S . We show that moving player m behind this specific player is actually more beneficial with respect to coalition $S \cup \{i\}$, which contradicts the optimality of the algorithm.

Denote the processing order when player m is considered by the algorithm with respect to coalition S by τ^S and with respect to coalition $S \cup \{i\}$ by $\tau^{S \cup \{i\}}$. Let r^S denote the player where player m will be moved behind with respect to coalition S according to the algorithm. Similarly, let $r^{S \cup \{i\}}$ denote the player where player m will be moved behind with respect to coalition $S \cup \{i\}$ according to the algorithm. Note that in case player m is

not moved by the algorithm with respect to coalition $S \cup \{i\}$, then we define player $r^{S \cup \{i\}}$ as player m . Since $c(m, \sigma_{S \cup \{i\}}) = c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})$ and $c(m, \sigma_S) = c(r^S, \tau^S)$, we have

$$c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}}) < c(r^S, \tau^S). \quad (15)$$

As we will see later $(S \cup \{i\})_{c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})}^{\sigma_0, \tau^S} \neq \emptyset$, let $\tilde{r}^S \in (S \cup \{i\})_{c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})}^{\sigma_0, \tau^S}$ be such that player m would be moved behind this player in case player m is moved to component $(S \cup \{i\})_{c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})}^{\sigma_0, \tau^S}$ according to the algorithm with respect to coalition S . Note that player \tilde{r}^S is unique because the algorithm always selects a unique player per component. Note that player \tilde{r}^S might also be player i as in this way we make sure that player \tilde{r}^S also exists if $c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}}) = c(i, \sigma_0)$.³ Note that in case player m and $r^{S \cup \{i\}}$ coincide (that means that player m is not moved by the algorithm with respect to coalition $S \cup \{i\}$), then we define player \tilde{r}^S as player m .

Since player m is moved behind player r^S and not behind player \tilde{r}^S , we have due to property (v) of the algorithm that

$$\alpha_{S \tau^S(\tilde{r}^S, r^S]} p_m - \alpha_m p_{N \tau^S(\tilde{r}^S, r^S]} > 0. \quad (16)$$

We distinguish between two cases:

- case A: $\{k \in S \mid \sigma_0^S(m) < \sigma_0^S(k) < \sigma_0^S(i)\} = \emptyset$,
- case B: $\{k \in S \mid \sigma_0^S(m) < \sigma_0^S(k) < \sigma_0^S(i)\} \neq \emptyset$.

Case A: $[\{k \in S \mid \sigma_0^S(m) < \sigma_0^S(k) < \sigma_0^S(i)\} \neq \emptyset]$

Hence, there are no players of coalition S in between player m and player i in σ_0^S . Then, from (14) it follows that for every $k \in S \cap F(\tau^S, m)$ we have

$$c(k, \tau^S) = c(k, \tau^{S \cup \{i\}}). \quad (17)$$

Note that this implies that $(S \cup \{i\})_{c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})}^{\sigma_0, \tau^S}$ is non-empty.

We will prove that moving player m behind player r^S is more beneficial than moving player m behind player $r^{S \cup \{i\}}$, i.e., we will prove that

$$\alpha_{(S \cup \{i\}) \tau^{S \cup \{i\}}(r^{S \cup \{i\}}, r^S]} p_m - \alpha_m p_{N \tau^{S \cup \{i\}}(r^{S \cup \{i\}}, r^S]} > 0. \quad (18)$$

This would imply that the step made by the algorithm for player m when applied on coalition $S \cup \{i\}$ is not optimal, which contradicts the optimality of the algorithm. Hence, for case A, it remains to prove (18).

We distinguish from now on between the following four cases:

- case A.1: $i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]$,
- case A.2: $i \in N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]$,

³Note that in case $c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}}) = c(i, \sigma_0)$ it is not admissible for the algorithm to move player m to component $(S \cup \{i\})_{c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})}^{\sigma_0, \tau^S}$ due to requirement (ii) of admissibility, but this is no problem as also in this case the upcoming arguments are still valid.

- case A.3: $i \in N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \in N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]$,
- case A.4: $i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \in N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]$.

Case A.1: $[i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]$

Note that this case occurs if $r^{S \cup \{i\}} \neq m$ and player i is not necessarily moved by the algorithm with respect to coalition $S \cup \{i\}$. Then, it follows from (17) together with the fact $c(\tilde{r}^S, \tau^S) = c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})$ that

$$\alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]} p_m - \alpha_m p_{N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]} = \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S, r^S]} \stackrel{(16)}{>} 0.$$

Case A.2: $[i \in N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]$

Note that this case occurs if $r^{S \cup \{i\}} = m$ and player i has been moved by the algorithm with respect to coalition $S \cup \{i\}$ such that $\tau^{S \cup \{i\}}(i) > \tau^{S \cup \{i\}}(r^S)$. Then, using the same arguments as in case A.1, we have

$$\alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]} p_m - \alpha_m p_{N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]} = \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S, r^S]} + \alpha_m p_i \stackrel{(16)}{>} 0.$$

Case A.3: $[i \in N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \in N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]$

Note that this case occurs if $r^{S \cup \{i\}} = m$ and player i has either not been moved by the algorithm with respect to coalition $S \cup \{i\}$ or it has been moved such that $\tau^{S \cup \{i\}}(i) < \tau^{S \cup \{i\}}(r^S)$. Then, using the same arguments as in case A.1, we have

$$\alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]} p_m - \alpha_m p_{N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]} = \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S, r^S]} + \alpha_i p_m \stackrel{(16)}{>} 0.$$

Case A.4: $[i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \in N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]$

Note that this case occurs if $r^{S \cup \{i\}} \neq m$ and player i has been moved by the algorithm with respect to coalition $S \cup \{i\}$ such that $\tau^{S \cup \{i\}}(r^{S \cup \{i\}}) < \tau^{S \cup \{i\}}(i) < \tau^{S \cup \{i\}}(r^S)$. Then, using the same arguments as in case A.1, we have

$$\alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]} p_m - \alpha_m p_{N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]} = \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S, r^S]} + \alpha_i p_m - \alpha_m p_i.$$

If we can show that $\alpha_i p_m - \alpha_m p_i > 0$, (18) follows.

Let \hat{r}^S be the direct predecessor of player i in $\tau^{S \cup \{i\}}$. Since player i is moved behind player \hat{r}^S and not behind player r^S , we have due to the optimality of the algorithm that

$$\alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(i, r^S]} p_i - \alpha_i p_{N^{\tau^{S \cup \{i\}}}(i, r^S]} \leq 0.$$

Consequently, it follows from (17) together with the fact $c(\hat{r}^S, \tau^S) = c(i, \tau^{S \cup \{i\}})$ that we also have

$$\alpha_{S^{\tau^S}(\hat{r}^S, r^S]} p_i - \alpha_i p_{N^{\tau^S}(\hat{r}^S, r^S]} \leq 0.$$

Therefore, together with (16), we can conclude $\frac{\alpha_i}{p_i} > \frac{\alpha_m}{p_m}$, i.e., $\alpha_i p_m - \alpha_m p_i > 0$.

Case B: $[\{k \in S \mid \sigma_0^S(m) < \sigma_0^S(k) < \sigma_0^S(i)\} \neq \emptyset]$

Hence, there are players of coalition S in between player m and player i in σ_0^S . Therefore,

due to the definition of player m , it follows that for every $k \in S$ with $\sigma_0^S(m) < \sigma_0^S(k) < \sigma_0^S(i)$ we have

$$c(k, \sigma_{S \cup \{i\}}) \geq c(k, \sigma_S), \quad (19)$$

i.e., the statement in the lemma holds for all followers of player m with respect to σ_0^S intersected with $S \cap P(\sigma_0^S, i)$. First, note that $(S \cup \{i\})_{c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})}^{\sigma_0, \tau^S}$ is non-empty because of the following. Due to requirement (i) and (ii) of admissibility we know that the first player in $(S \cup \{i\})_{c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})}^{\sigma_0, \tau^{S \cup \{i\}}}$ with respect to $\tau^{S \cup \{i\}}$ is also the first player in $(S \cup \{i\})_{c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})}^{\sigma_0, \sigma_0^S}$ with respect to σ_0^S . Therefore, using (19) we know that this player also belongs to $(S \cup \{i\})_{c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})}^{\sigma_0, \tau^S}$. Hence, $(S \cup \{i\})_{c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})}^{\sigma_0, \tau^S}$ is non-empty.

Next, define player $l \in S^{\tau^S}(\tilde{r}^S, r^S]$ as the player in $S^{\tau^S}(\tilde{r}^S, r^S]$ who is positioned last with respect to $\tau^{S \cup \{i\}}$, i.e., $\tau^{S \cup \{i\}}(l) \geq \tau^{S \cup \{i\}}(k)$ for all $k \in S^{\tau^S}(\tilde{r}^S, r^S]$. Note that player l was actually player r^S in case A because of (17). From the assumptions in (19) and (15) it follows that

$$c(l, \tau^{S \cup \{i\}}) \geq c(r^S, \tau^{S \cup \{i\}}) \geq c(r^S, \tau^S) > c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}}), \quad (20)$$

i.e., player l is to the right of player $r^{S \cup \{i\}}$ in $\tau^{S \cup \{i\}}$. We will prove that moving player m behind player l is more beneficial than moving player m behind player $r^{S \cup \{i\}}$, i.e., we will prove that

$$\alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} p_m - \alpha_m p_{N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} > 0. \quad (21)$$

This implies that the step made by the algorithm for player m when applied on coalition $S \cup \{i\}$ is not optimal, which contradicts the optimality of the algorithm. Hence, for case B, it remains to prove (21).

From the definitions of players \tilde{r}^S and l , together with (19), it follows that

$$S^{\tau^S}(\tilde{r}^S, r^S] \subseteq S^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \subseteq (S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]. \quad (22)$$

Moreover, since during the run of the algorithm player m jumps over all players in $S^{\tau^S}(m, r^S]$, Proposition (3.1) implies that

$$u_k > u_m, \quad (23)$$

for all $k \in S^{\tau^S}(m, r^S]$.

Below we distinguish between the following four cases:

- case B.1: $i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]$,
- case B.2: $i \in N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]$,
- case B.3: $i \in N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \in N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]$,
- case B.4: $i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \in N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]$.

Case B.1: $[i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]$

Using $c(l, \tau^{S \cup \{i\}}) \geq c(r^S, \tau^S)$ from (20), we distinguish between another two cases:

- case B.1(i): $c(l, \tau^{S \cup \{i\}}) = c(r^S, \tau^S)$,

– case B.1(ii): $c(l, \tau^{S \cup \{i\}}) > c(r^S, \tau^S)$.

Case B.1(i): $[c(l, \tau^{S \cup \{i\}}) = c(r^S, \tau^S)]$

Since $\tilde{r}^S \in (S \cup \{i\})_{c(r^S, \tau^S)}^{\sigma_0, \tau^{S \cup \{i\}}}$ and thus $c(\tilde{r}^S, \tau^S) = c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})$, the assumptions $i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $c(l, \tau^{S \cup \{i\}}) = c(r^S, \tau^S)$ imply that

$$\overline{S^{\tau^S}}(\tilde{r}^S, r^S] = \overline{S^{\tau^{S \cup \{i\}}}}(r^{S \cup \{i\}}, l] = (\overline{S \cup \{i\}})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]. \quad (24)$$

For every $k \in S^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus S^{\tau^S}(\tilde{r}^S, r^S]$ we know that $k \in S^{\tau^S}(m, \tilde{r}^S]$, because $c(k, \tau^S) \leq c(k, \tau^{S \cup \{i\}})$ by (19) and $P(\tau^S, m) = P(\tau^{S \cup \{i\}}, m)$. Hence, also $k \in S^{\tau^S}(m, r^S]$ and thus from (23) it follows that $u_k > u_m$. Together with Proposition 3.2 applied on the set $S^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus S^{\tau^S}(\tilde{r}^S, r^S]$ and player m we have

$$\begin{aligned} & \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus S^{\tau^S}(\tilde{r}^S, r^S]} \\ &= \alpha_{S^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{S^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus S^{\tau^S}(\tilde{r}^S, r^S]} > 0, \end{aligned} \quad (25)$$

where the equality follows from the assumption $i \notin N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]$. As a consequence,

$$\begin{aligned} & \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] p_m - \alpha_m p_{N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} \\ &= \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] p_m - \alpha_m \left(p_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} + p_{\overline{(S \cup \{i\})}^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} \right) \\ &\stackrel{(24)}{=} \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] p_m - \alpha_m \left(p_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} + p_{\overline{S^{\tau^S}}(\tilde{r}^S, r^S]} \right) \\ &\stackrel{(22)}{=} \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m \left(p_{S^{\tau^S}(\tilde{r}^S, r^S]} + p_{\overline{S^{\tau^S}}(\tilde{r}^S, r^S]} \right) \\ &\quad + \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus S^{\tau^S}(\tilde{r}^S, r^S]} \\ &\stackrel{(25)}{\geq} \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m \left(p_{S^{\tau^S}(\tilde{r}^S, r^S]} + p_{\overline{S^{\tau^S}}(\tilde{r}^S, r^S]} \right) \\ &\stackrel{(16)}{>} 0, \end{aligned}$$

proving (21).

Case B.1(ii): $[c(l, \tau^{S \cup \{i\}}) > c(r^S, \tau^S)]$

Define Q as the set of players from $S^{\tau^S}(\tilde{r}^S, r^S]$ who are positioned in $\tau^{S \cup \{i\}}$ in a component to the right of component $(S \cup \{i\})_{c(r^S, \tau^S)}^{\sigma_0, \tau^{S \cup \{i\}}}$. Since $c(l, \tau^{S \cup \{i\}}) > c(r^S, \tau^S)$ and $l \in S^{\tau^S}(\tilde{r}^S, r^S]$, we have $l \in Q$ and thus $Q \neq \emptyset$. Select from Q only the players who have in their corresponding component in $\tau^{S \cup \{i\}}$ no players from $S^{\tau^S}(\tilde{r}^S, r^S]$ in front of him and denote this set of players by \mathcal{Q} , i.e., we select in each component a player of Q (if possible) that is most to the left in $\tau^{S \cup \{i\}}$.

Set $\mathcal{Q} = \{q_1, q_2, \dots, q_{|\mathcal{Q}|}\}$ such that $c(q_k, \tau^{S \cup \{i\}}) < c(q_{k+1}, \tau^{S \cup \{i\}})$ for all $k \in \{1, \dots, |\mathcal{Q}| - 1\}$. Define w_1 as the first player in $\overline{(S \cup \{i\})}_{c(r^S, \tau^S)}^{\sigma_0}$, i.e., the direct follower in $\tau^{S \cup \{i\}}$ of component $(S \cup \{i\})_{c(r^S, \tau^S)}^{\sigma_0, \tau^{S \cup \{i\}}}$. For $k \in \{2, \dots, |\mathcal{Q}|\}$, define w_k as the first player in $\overline{(S \cup \{i\})}_{c(q_{k-1}, \tau^{S \cup \{i\}})}^{\sigma_0}$, i.e., the direct follower in $\tau^{S \cup \{i\}}$ of component $(S \cup \{i\})_{c(q_{k-1}, \tau^{S \cup \{i\}})}^{\sigma_0, \tau^{S \cup \{i\}}}$. Note that because of the definition of player l we have $c(q_{|\mathcal{Q}|}, \tau^{S \cup \{i\}}) = c(l, \tau^{S \cup \{i\}})$.

The collection of sets $\{N^{\tau^{S \cup \{i\}}}[w_k, q_k] \mid k \in \{1, \dots, |\mathcal{Q}|\}\}$ are by definition mutually disjoint. Moreover, for $k \in \{1, \dots, |\mathcal{Q}|\}$, we have $N^{\tau^{S \cup \{i\}}}[w_k, q_k] \cap S^{\tau^S}(\tilde{r}^S, r^S] =$

\emptyset and $(S \cup \{i\})^{\tau^{S \cup \{i\}}}[w_k, q_k) \subseteq (S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]$. If we set $R = S^{\tau^S}(\tilde{r}^S, r^S] \cup \bigcup_{k=1}^{|\mathcal{Q}|} (S \cup \{i\})^{\tau^{S \cup \{i\}}}[w_k, q_k)$, then using (22) we have

$$R \subseteq (S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l], \quad (26)$$

and

$$\alpha_R = \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} + \sum_{k=1}^{|\mathcal{Q}|} \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}[w_k, q_k)}, \quad (27)$$

and

$$p_R = p_{S^{\tau^S}(\tilde{r}^S, r^S]} + \sum_{k=1}^{|\mathcal{Q}|} p_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}[w_k, q_k)}. \quad (28)$$

Note, for $k \in \{1, \dots, |\mathcal{Q}|\}$, we have $c(q_k, \sigma_0) \leq c(q_k, \tau^S) \leq c(r^S, \tau^S)$. Hence, as w_k is in $\tau^{S \cup \{i\}}$ to the right of component $(S \cup \{i\})_{c(r^S, \tau^S)}^{\sigma_0, \tau^{S \cup \{i\}}}$, we have that it is admissible to move player q_k in front of w_k with respect to $\tau^{S \cup \{i\}}$. In other words, it is admissible to move player q_k to the tail of component $(S \cup \{i\})_{c(q_{k-1}, \tau^{S \cup \{i\}})}^{\sigma_0, \tau^{S \cup \{i\}}}$ (and the tail of component $(S \cup \{i\})_{c(r^S, \tau^S)}^{\sigma_0, \tau^{S \cup \{i\}}}$ in case $k = 1$). Since player q_k is not moved behind player w_k , we have due to property (v) of the algorithm that

$$\alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}[w_k, q_k)} p_{q_k} - \alpha_{q_k} p_{N^{\tau^{S \cup \{i\}}}[w_k, q_k)} > 0.$$

Moreover, since $q_k \in S^{\tau^S}(m, r^S]$ it follows from (23) that $u_m < u_{q_k}$. As a consequence,

$$\alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}[w_k, q_k)} p_m - \alpha_m p_{N^{\tau^{S \cup \{i\}}}[w_k, q_k)} > 0. \quad (29)$$

We have

$$\begin{aligned} & \alpha_R p_m - \alpha_m \left(p_R + p_{(\overline{S \cup \{i\}})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} \right) \\ \stackrel{(27), (28)}{=} & \left(\alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} + \sum_{k=1}^{|\mathcal{Q}|} \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}[w_k, q_k)} \right) p_m \\ & - \alpha_m \left(p_{S^{\tau^S}(\tilde{r}^S, r^S]} + \sum_{k=1}^{|\mathcal{Q}|} p_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}[w_k, q_k)} + p_{(\overline{S \cup \{i\}})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, w_1)} + p_{(\overline{S \cup \{i\}})^{\tau^{S \cup \{i\}}}[w_1, l]} \right) \\ \stackrel{\text{see below}}{=} & \left(\alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} + \sum_{k=1}^{|\mathcal{Q}|} \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}[w_k, q_k)} \right) p_m \\ & - \alpha_m \left(p_{S^{\tau^S}(\tilde{r}^S, r^S]} + \sum_{k=1}^{|\mathcal{Q}|} p_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}[w_k, q_k)} + p_{\overline{S^{\tau^S}(\tilde{r}^S, r^S]}} + \sum_{k=1}^{|\mathcal{Q}|} p_{(\overline{S \cup \{i\}})^{\tau^{S \cup \{i\}}}[w_k, q_k)} \right) \\ = & \left(\alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} + \sum_{k=1}^{|\mathcal{Q}|} \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}[w_k, q_k)} \right) p_m - \alpha_m \left(p_{N^{\tau^S}(\tilde{r}^S, r^S]} + \sum_{k=1}^{|\mathcal{Q}|} p_{N^{\tau^{S \cup \{i\}}}[w_k, q_k)} \right) \\ \stackrel{(16), (29)}{>} & 0. \end{aligned} \quad (30)$$

Note that the second equality follows from $c(\tilde{r}^S, \tau^S) = c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})$, the fact that w_1 is the direct follower in $\tau^{S \cup \{i\}}$ of component $(S \cup \{i\})_{c(r^S, \tau^S)}^{\sigma_0, \tau^{S \cup \{i\}}}$, and the assumption

$i \notin N^{\tau^S}(\tilde{r}^S, r^S]$. Moreover, the collection of sets $\{(\overline{S \cup \{i\}})^{\tau^{S \cup \{i\}}}[w_k, q_k] \mid k \in \{1, \dots, |\mathcal{Q}|\}\}$ forms a partition of the set $(\overline{S \cup \{i\}})^{\tau^{S \cup \{i\}}}[w_1, l]$.

For every $k \in S^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus R$, we know that either $k \in S^{\tau^S}(m, \tilde{r}^S]$ or $c(k, \tau^S) \geq c(r^S, \tau^S)$. If $k \in S^{\tau^S}(m, \tilde{r}^S]$, then also $k \in S^{\tau^S}(m, r^S]$ and thus from (23) we have $u_k > u_m$. Next, if $c(k, \tau^S) \geq c(r^S, \tau^S)$, then because $c(l, \tau^S) \leq c(r^S, \tau^S)$ we know that the swap of players k and l with respect to $\tau^{S \cup \{i\}}$ is admissible. Therefore, according to Proposition 3.1, we know $u_k \geq u_l$. Moreover, since $l \in S^{\tau^S}(m, r^S]$, it follows from (23) that $u_m < u_l$. As a consequence, $u_m < u_k$. Together with Proposition 3.2 applied on the set $S^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus R$ and player m we have

$$\begin{aligned} & \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus R} p_m - \alpha_m p_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus R} \\ &= \alpha_{S^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus R} p_m - \alpha_m p_{S^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus R} > 0, \end{aligned} \quad (31)$$

where the equality follows from the assumption $i \notin N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]$. As a consequence,

$$\begin{aligned} & \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} p_m - \alpha_m p_{N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} \\ &= \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} p_m - \alpha_m \left(p_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} + p_{(\overline{S \cup \{i\}})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} \right) \\ &\stackrel{(26)}{=} \alpha_R p_m - \alpha_m \left(p_R + p_{(\overline{S \cup \{i\}})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} \right) \\ &\quad + \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus R} p_m - \alpha_m p_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus R} \\ &\stackrel{(30), (31)}{>} 0, \end{aligned}$$

proving (21).

Case B.2: $[i \in N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]$

Note that this case occurs exactly if $c(\tilde{r}^S, \tau^S) < c(i, \sigma_0) < c(r^S, \tau^S)$ and player i has been moved by the algorithm with respect to coalition $S \cup \{i\}$ such that $\tau^{S \cup \{i\}}(i) > \tau^{S \cup \{i\}}(l)$. We have

$$\alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S, r^S] \setminus \{i\}} = \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S, r^S]} + \alpha_m p_i \stackrel{(16)}{>} 0. \quad (32)$$

Then, using the same arguments as in case B.1, we can prove (21). Namely, where we used (16) in case B.1, we now use the above equation. Hence, player i has already been taken into account and thus for using the same arguments as in case B.1 we can assume $i \notin N^{\tau^S}(\tilde{r}^S, r^S]$ and $i \notin N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]$.

For example, analogous to case B.1(i), case B.2(i) goes as follows. Since $\tilde{r}^S \in (S \cup \{i\})_{c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})}^{\sigma_0, \tau^S}$ and thus $c(\tilde{r}^S, \tau^S) = c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}})$, the assumption $c(l, \tau^{S \cup \{i\}}) = c(r^S, \tau^S)$ implies that

$$\overline{S^{\tau^S}(\tilde{r}^S, r^S] \setminus \{i\}} = \overline{S^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} = (\overline{S \cup \{i\}})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]. \quad (33)$$

Using exactly the same arguments as in case B.1(i) we have

$$\alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus S^{\tau^S}(\tilde{r}^S, r^S]} > 0, \quad (34)$$

As a consequence,

$$\alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} p_m - \alpha_m p_{N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]}$$

$$\begin{aligned}
&= \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} p_m - \alpha_m \left(p_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} + p_{(\overline{S \cup \{i\}})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} \right) \\
&\stackrel{(33)}{=} \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} p_m - \alpha_m \left(p_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]} + p_{\overline{S^{\tau^S}}(\tilde{r}^S, r^S) \setminus \{i\}} \right) \\
&\stackrel{(22)}{=} \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m \left(p_{S^{\tau^S}(\tilde{r}^S, r^S]} + p_{\overline{S^{\tau^S}}(\tilde{r}^S, r^S) \setminus \{i\}} \right) \\
&\quad + \alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l] \setminus S^{\tau^S}(\tilde{r}^S, r^S]} \\
&\stackrel{(34)}{\geq} \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m \left(p_{S^{\tau^S}(\tilde{r}^S, r^S]} + p_{\overline{S^{\tau^S}}(\tilde{r}^S, r^S) \setminus \{i\}} \right) \\
&= \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S, r^S) \setminus \{i\}} \\
&\stackrel{(32)}{>} 0,
\end{aligned}$$

proving (21).

Case B.3: $[i \in N^{\tau^S}(\tilde{r}^S, r^S)]$ and $i \in N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]$

Note that this case occurs exactly if $c(\tilde{r}^S, \tau^S) < c(i, \sigma_0) < c(r^S, \tau^S)$ and player i has either not been moved by the algorithm with respect to coalition $S \cup \{i\}$ or it has been moved such that $\tau^{S \cup \{i\}}(i) < \tau^{S \cup \{i\}}(l)$. We have

$$\alpha_{S^{\tau^S}(\tilde{r}^S, r^S) \cup \{i\}} p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S, r^S)} = \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S, r^S]} + \alpha_i p_m \stackrel{(16)}{>} 0.$$

Then, using the same arguments as in case B.1, we can prove (21).

Case B.4: $[i \notin N^{\tau^S}(\tilde{r}^S, r^S)]$ and $i \in N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, l]$

Then,

$$\alpha_{S^{\tau^S}(\tilde{r}^S, r^S) \cup \{i\}} p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S, r^S) \cup \{i\}} = \alpha_{S^{\tau^S}(\tilde{r}^S, r^S]} p_m - \alpha_m p_{N^{\tau^S}(\tilde{r}^S, r^S]} + \alpha_i p_m - \alpha_m p_i.$$

It suffices to show that $\alpha_i p_m - \alpha_m p_i > 0$. Together with the above equation and (16), we can prove (21) by using the same arguments as in case B.1.

We distinguish between two cases:

- case B.4(i): $c(i, \sigma_0) > c(r^S, \tau^S)$,
- case B.4(ii): $c(i, \sigma_0) < c(\tilde{r}^S, \tau^S)$.

Case B.4(i): $[c(i, \sigma_0) > c(r^S, \tau^S)]$

Note that this case occurs exactly if $\tau^{S \cup \{i\}}(i) < \tau^{S \cup \{i\}}(l)$. This means that player i is not necessarily moved by the algorithm with respect to coalition $S \cup \{i\}$. Then,

$$c(l, \sigma_0) \leq c(l, \tau^S) \leq c(r^S, \tau^S) < c(i, \sigma_0) \leq c(i, \tau^{S \cup \{i\}}) = c(i, \sigma_{S \cup \{i\}}).$$

Hence, the swap of players i and l with respect to $\sigma_{S \cup \{i\}}$ is admissible and thus, according to Proposition 3.1, we have $u_i \geq u_l$. Consequently, together with (23), we have $u_i \geq u_l > u_m$. Hence, $\alpha_i p_m - \alpha_m p_i > 0$.

Case B.4(ii): $[c(i, \sigma_0) < c(\tilde{r}^S, \tau^S)]$

Note that this case occurs exactly if player i has been moved by the algorithm with

respect to coalition $S \cup \{i\}$ such that $\tau^{S \cup \{i\}}(r^{S \cup \{i\}}) < \tau^{S \cup \{i\}}(i) < \tau^{S \cup \{i\}}(l)$. Suppose $c(l, \sigma_0) < c(i, \sigma_0)$, then

$$c(l, \sigma_0) < c(i, \sigma_0) < c(\tilde{r}^S, \tau^S) = c(r^{S \cup \{i\}}, \tau^{S \cup \{i\}}) = c(m, \sigma_{S \cup \{i\}}).$$

Hence, the swap of players m and l with respect to $\sigma_{S \cup \{i\}}$ is admissible and thus, according to Proposition 3.1, we have $u_m \geq u_l$. This is a contradiction with (23) and thus we know $c(l, \sigma_0) > c(i, \sigma_0)$. Therefore, using (14), we have $c(l, \tau^S) = c(l, \tau^{S \cup \{i\}})$ which implies $l = r^S$. As a consequence, if $k \in S^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S]$, then also $k \in S^{\tau^S}(m, r^S]$ and thus from (23) we have $u_m < u_k$. With the same arguments we used for player l we can conclude $c(k, \sigma_0) > c(i, \sigma_0)$ and thus $c(k, \tau^S) = c(k, \tau^{S \cup \{i\}})$. Therefore, we have

$$N^{\tau^S}(\tilde{r}^S, r^S] \cup \{i\} = N^{\tau^{S \cup \{i\}}}(r^{S \cup \{i\}}, r^S], \quad (35)$$

and

$$c(k, \tau^S) = c(k, \tau^{S \cup \{i\}}), \quad (36)$$

for all $k \in S^{\tau^S}(\tilde{r}^S, r^S]$.

Let \hat{r}^S be the direct predecessor of player i in $\tau^{S \cup \{i\}}$. Since player i is in $\tau^{S \cup \{i\}}$ behind player \hat{r}^S and not behind player r^S , although this is an admissible swap, we have

$$\alpha_{(S \cup \{i\})^{\tau^{S \cup \{i\}}}(i, r^S]} p_i - \alpha_i p_{N^{\tau^{S \cup \{i\}}}(i, r^S]} \leq 0.$$

Consequently, it follows from (35) and (36) together with the fact that $c(\hat{r}^S, \tau^S) = c(i, \tau^{S \cup \{i\}})$, that

$$\alpha_{S^{\tau^S}(\hat{r}^S, r^S]} p_i - \alpha_i p_{N^{\tau^S}(\hat{r}^S, r^S]} \leq 0.$$

Moreover, since player m is moved behind player r^S and not behind player \hat{r}^S , we have due to property (v) of the algorithm that

$$\alpha_{S^{\tau^S}(\hat{r}^S, r^S]} p_m - \alpha_m p_{N^{\tau^S}(\hat{r}^S, r^S]} > 0.$$

Therefore, we can conclude $\frac{\alpha_i}{p_i} > \frac{\alpha_m}{p_m}$, i.e., $\alpha_i p_m - \alpha_m p_i > 0$, which is exactly what we needed.

B Proof of Lemma 3.7

In this proof we denote $c(k, S \cup \{i\}, \sigma)$ by $c(k, \sigma)$ for every $k \in S \cup \{i\}$ and every $\sigma \in \Pi(N)$. We continue the proof by means of induction on the number

$$n_m = \{j \in S \mid \sigma_S(m) < \sigma_S(j) < \sigma_S(i)\},$$

i.e., the number of players in coalition S between player m and player i in σ_S .

Base step: If $n_m = 0$, then player m is among all players in S the closest predecessor of player i with respect to σ_S . Then,

$$\{l \in S \cap F(\tau_m, m) \mid c(l, \tau_m) < c(i, \sigma_0)\} = \emptyset,$$

and thus (4) is true.

Induction step: Assume that (4) holds for every $k \in S \cap P(\sigma_S, i)$ with $n_k < n_m$. Since the followers of player m with respect to σ_S intersected with $S \cap P(\sigma_S, i)$ are exactly the players with $n_k < n_m$, we actually assume that for every $k \in S \cap P(\sigma_S, i) \cap F(\sigma_S, m)$ and $r \in S \cap F(\tau_k, k)$ with $c(r, \tau_k) < c(i, \sigma_0)$ we have

$$\alpha_{(S \cup \{i\})^{\tau_k}(k, r]} p_k - \alpha_k p_{N^{\tau_k}(k, r]} \leq 0, \quad (37)$$

We distinguish between the following cases:

- $u_m \leq u_k$ for all $k \in S^{\sigma_S}(m, l]$,
- there exists a player $k \in S^{\sigma_S}(m, l]$ with $u_m > u_k$.

Case 1: [$u_m \leq u_k$ for all $k \in S^{\sigma_S}(m, l]$]

We will show that since moving player m behind player l in σ_S is not beneficial, also moving player m behind player l in τ_m is not beneficial.

Note that because of the induction assumption in (37), we know that all players in $(S \cup \{i\})^{\tau_m}(m, l]$ have not been moved by the algorithm and thus $N^{\tau_m}(m, l] \subseteq N^{\sigma_S}(m, l]$. Moreover, since player l is in both σ_S and τ_m to the left of the original component of player i in σ_0 , we have

$$(S \cup \{i\})^{\tau_m}(m, l] \subseteq S^{\sigma_S}(m, l], \quad (38)$$

and

$$(\overline{S \cup \{i\}})^{\tau_m}(m, l] = \overline{S}^{\sigma_S}(m, l]. \quad (39)$$

Note, as processing order σ_S is optimal for coalition S , we have

$$\alpha_{S^{\sigma_S}(m, l]} p_m - \alpha_m p_{N^{\sigma_S}(m, l]} \leq 0. \quad (40)$$

Since $u_m \leq u_k$ for all $k \in S^{\sigma_S}(m, l]$, it follows from Proposition 3.2 applied on the set $S^{\sigma_S}(m, l]$ and player m together with (38) that

$$\alpha_{S^{\sigma_S}(m, l] \setminus (S \cup \{i\})^{\tau_m}(m, l]} p_m - \alpha_m p_{S^{\sigma_S}(m, l] \setminus (S \cup \{i\})^{\tau_m}(m, l]} \geq 0, \quad (41)$$

where there is an equality if $S^{\sigma_S}(m, l] = (S \cup \{i\})^{\tau_m}(m, l]$. As a consequence, we have

$$\begin{aligned} & \alpha_{(S \cup \{i\})^{\tau_m}(m, l]} p_m - \alpha_m p_{N^{\tau_m}(m, l]} \\ = & \alpha_{(S \cup \{i\})^{\tau_m}(m, l]} p_m - \alpha_m \left(p_{(S \cup \{i\})^{\tau_m}(m, l]} + p_{\overline{(S \cup \{i\})^{\tau_m}(m, l]}}^{\tau_m}(m, l] \right) \\ \stackrel{(38), (39)}{=} & \alpha_{S^{\sigma_S}(m, l]} p_m - \alpha_m \left(p_{S^{\sigma_S}(m, l]} + p_{\overline{S}^{\sigma_S}(m, l]} \right) \\ & - \left(\alpha_{S^{\sigma_S}(m, l] \setminus (S \cup \{i\})^{\tau_m}(m, l]} p_m - \alpha_m p_{S^{\sigma_S}(m, l] \setminus (S \cup \{i\})^{\tau_m}(m, l]} \right) \\ = & \alpha_{S^{\sigma_S}(m, l]} p_m - \alpha_m p_{N^{\sigma_S}(m, l]} \\ & - \left(\alpha_{S^{\sigma_S}(m, l] \setminus (S \cup \{i\})^{\tau_m}(m, l]} p_m - \alpha_m p_{S^{\sigma_S}(m, l] \setminus (S \cup \{i\})^{\tau_m}(m, l]} \right) \\ \stackrel{(40), (41)}{\leq} & 0, \end{aligned}$$

and (4) follows.

Case 2: [there exists a player $k \in S^{\sigma_S}(m, l]$ with $u_m > u_k$]

Let player $q \in S^{\sigma_S}(m, l]$ be the closest follower of player m in σ_S with a smaller urgency

than player m : $u_q < u_m \leq u_k$ for all $k \in S^{\sigma_S}(m, q)$. Since $q \in S \cap P(\sigma_S, i) \cap F(\sigma_S, m)$, it follows from the induction assumption in (37) that

$$\alpha_{(S \cup \{i\})^{\tau_q}(q, l]} p_q - \alpha_q p_{N^{\tau_q}(q, l]} \leq 0.$$

As a consequence, since $u_q < u_m$, we also have

$$\alpha_{(S \cup \{i\})^{\tau_q}(q, l]} p_m - \alpha_m p_{N^{\tau_q}(q, l]} < 0. \quad (42)$$

We distinguish between two cases:

- case 2(i): $S^{\sigma_S}(m, q) \cap S^{\tau_m}(m, l] = \emptyset$, i.e., all players in $S^{\sigma_S}(m, q)$ have been moved by the algorithm,
- case 2(ii): $S^{\sigma_S}(m, q) \cap S^{\tau_m}(m, l] \neq \emptyset$, i.e., not all players in $S^{\sigma_S}(m, q)$ have been moved by the algorithm.

Case 2(i): $[S^{\sigma_S}(m, q) \cap S^{\tau_m}(m, l] = \emptyset]$

We will show that since moving player q behind player l in τ_q is not beneficial, also moving player m behind player l in τ_m is not beneficial.

Since player m is a predecessor of player q in σ_S , we have

$$(\overline{S \cup \{i\}})^{\tau_q}(q, l] = (\overline{S \cup \{i\}})^{\sigma_S}(q, l] \subseteq (\overline{S \cup \{i\}})^{\sigma_S}(m, l] = (\overline{S \cup \{i\}})^{\tau_m}(m, l], \quad (43)$$

with an equality in case $c(q, \sigma_S) = c(m, \sigma_S)$. Since all players in $S^{\sigma_S}(m, q)$ have been moved by the algorithm, we have

$$(S \cup \{i\})^{\tau_m}(m, l] = (S \cup \{i\})^{\tau_q}(q, l], \quad (44)$$

if player q has also been moved by the algorithm, and

$$(S \cup \{i\})^{\tau_m}(m, l] = (S \cup \{i\})^{\tau_q}(q, l] \cup \{q\}, \quad (45)$$

if player q has not been moved by the algorithm. As a consequence,

$$\begin{aligned} & \alpha_{(S \cup \{i\})^{\tau_m}(m, l]} p_m - \alpha_m p_{N^{\tau_m}(m, l]} \\ &= \alpha_{(S \cup \{i\})^{\tau_m}(m, l]} p_m - \alpha_m (p_{(S \cup \{i\})^{\tau_m}(m, l]} + p_{(\overline{S \cup \{i\}})^{\tau_m}(m, l]}) \\ &\stackrel{(43), (44)}{=} \alpha_{(S \cup \{i\})^{\tau_q}(q, l]} p_m - \alpha_m (p_{(S \cup \{i\})^{\tau_q}(q, l]} + p_{(\overline{S \cup \{i\}})^{\tau_q}(q, l]} + p_{(\overline{S \cup \{i\}})^{\tau_m}(m, l] \setminus (\overline{S \cup \{i\}})^{\tau_q}(q, l]}) \\ &\leq \alpha_{(S \cup \{i\})^{\tau_q}(q, l]} p_m - \alpha_m (p_{(S \cup \{i\})^{\tau_q}(q, l]} + p_{(\overline{S \cup \{i\}})^{\tau_q}(q, l]}) \\ &= \alpha_{(S \cup \{i\})^{\tau_q}(q, l]} p_m - \alpha_m p_{N^{\tau_q}(q, l]} \stackrel{(42)}{<} 0, \end{aligned}$$

if player q has been moved by the algorithm, and

$$\begin{aligned} & \alpha_{(S \cup \{i\})^{\tau_m}(m, l]} p_m - \alpha_m p_{N^{\tau_m}(m, l]} \\ &= \alpha_{(S \cup \{i\})^{\tau_m}(m, l]} p_m - \alpha_m (p_{(S \cup \{i\})^{\tau_m}(m, l]} + p_{(\overline{S \cup \{i\}})^{\tau_m}(m, l]}) \\ &\stackrel{(43), (45)}{=} \alpha_{(S \cup \{i\})^{\tau_q}(q, l]} p_m - \alpha_m (p_{(S \cup \{i\})^{\tau_q}(q, l]} + p_{(\overline{S \cup \{i\}})^{\tau_q}(q, l]} + p_{(\overline{S \cup \{i\}})^{\tau_m}(m, l] \setminus (\overline{S \cup \{i\}})^{\tau_q}(q, l]}) \\ &\quad + \alpha_q p_m - \alpha_m p_q \\ &\stackrel{u_q < u_m}{<} \alpha_{(S \cup \{i\})^{\tau_q}(q, l]} p_m - \alpha_m (p_{(S \cup \{i\})^{\tau_q}(q, l]} + p_{(\overline{S \cup \{i\}})^{\tau_q}(q, l]}) \end{aligned}$$

$$= \alpha_{(S \cup \{i\})^{\tau_q}(q,l]} p_m - \alpha_m p_{N^{\tau_q}(q,l]} \stackrel{(42)}{<} 0,$$

if player q has not been moved by the algorithm, which proves (4).

Case 2(ii): $[S^{\sigma_S}(m, q) \cap S^{\tau_m}(m, l) \neq \emptyset]$

We move player m behind player l in τ_m in two stages. In the first stage, player m will be moved behind a specific player t . Then, in the second stage, player m will be moved behind player l . Using similar arguments as in case 1 we can show that the move in the first stage is not beneficial and using similar arguments as in case 2(i) we can show that the move in the second stage is not beneficial. As a consequence, since cost differences have an additive structure and because the moves in both stages are not beneficial, moving player m behind player l in τ_m is not beneficial.

Let player $t \in S^{\sigma_S}(m, q)$ be the closest predecessor of player q in σ_S who is also a member of $S^{\tau_m}(m, l]$. Note that due to the assumption $S^{\sigma_S}(m, q) \cap S^{\tau_m}(m, l] \neq \emptyset$, player t exists. Because player t is a predecessor of player q in σ_S , we have $u_m \leq u_k$ for all $k \in S^{\sigma_S}(m, t]$. Therefore, using the same arguments as in case 1, we have

$$\alpha_{(S \cup \{i\})^{\tau_m}(m,t]} p_m - \alpha_m p_{N^{\tau_m}(m,t]} \leq 0. \quad (46)$$

Since player t is a predecessor of player q in σ_S and because $t \in S^{\tau_m}(m, l]$ (which means that player t has not been moved by the algorithm), we have

$$(\overline{S \cup \{i\}})^{\tau_q}(q, l] = (\overline{S \cup \{i\}})^{\sigma_S}(q, l] \subseteq (\overline{S \cup \{i\}})^{\sigma_S}(t, l] = (\overline{S \cup \{i\}})^{\tau_t}(t, l] = (\overline{S \cup \{i\}})^{\tau_m}(t, l], \quad (47)$$

with an equality in case $c(q, \sigma_S) = c(t, \sigma_S)$. Because of the definition of player t we have

$$(S \cup \{i\})^{\tau_m}(t, l] = (S \cup \{i\})^{\tau_q}(q, l], \quad (48)$$

if player q has been moved by the algorithm, and

$$(S \cup \{i\})^{\tau_m}(t, l] = (S \cup \{i\})^{\tau_q}(q, l] \cup \{q\}, \quad (49)$$

if player q has not been moved by the algorithm. As a consequence, by using the additive structure of cost differences, we have

$$\begin{aligned} & \alpha_{(S \cup \{i\})^{\tau_m}(m,l]} p_m - \alpha_m p_{N^{\tau_m}(m,l]} \\ = & \left(\alpha_{(S \cup \{i\})^{\tau_m}(m,t]} + \alpha_{(S \cup \{i\})^{\tau_m}(t,l]} \right) p_m - \alpha_m \left(p_{N^{\tau_m}(m,t]} + p_{N^{\tau_m}(t,l]} \right) \\ \stackrel{(46)}{\leq} & \alpha_{(S \cup \{i\})^{\tau_m}(t,l]} p_m - \alpha_m \left(p_{(S \cup \{i\})^{\tau_m}(t,l]} + p_{(\overline{S \cup \{i\}})^{\tau_m}(t,l]} \right) \\ \stackrel{(47),(48)}{=} & \alpha_{(S \cup \{i\})^{\tau_q}(q,l]} p_m - \alpha_m \left(p_{(S \cup \{i\})^{\tau_q}(q,l]} + p_{(\overline{S \cup \{i\}})^{\tau_q}(q,l]} + p_{(\overline{S \cup \{i\}})^{\tau_m}(t,l] \setminus (\overline{S \cup \{i\}})^{\tau_q}(q,l]} \right) \\ \leq & \alpha_{(S \cup \{i\})^{\tau_q}(q,l]} p_m - \alpha_m \left(p_{(S \cup \{i\})^{\tau_q}(q,l]} + p_{(\overline{S \cup \{i\}})^{\tau_q}(q,l]} \right) \\ = & \alpha_{(S \cup \{i\})^{\tau_q}(q,l]} p_m - \alpha_m p_{N^{\tau_q}(q,l]} \stackrel{(42)}{<} 0, \end{aligned}$$

if player q has been moved by the algorithm, and

$$\begin{aligned} & \alpha_{(S \cup \{i\})^{\tau_m}(m,l]} p_m - \alpha_m p_{N^{\tau_m}(m,l]} \\ = & \left(\alpha_{(S \cup \{i\})^{\tau_m}(m,t]} + \alpha_{(S \cup \{i\})^{\tau_m}(t,l]} \right) p_m - \alpha_m \left(p_{N^{\tau_m}(m,t]} + p_{N^{\tau_m}(t,l]} \right) \end{aligned}$$

$$\begin{aligned}
& \stackrel{(46)}{\leq} \alpha_{(S \cup \{i\})^{\tau_m}(t,l]} p_m - \alpha_m \left(p_{(S \cup \{i\})^{\tau_m}(t,l]} + p_{\overline{(S \cup \{i\})^{\tau_m}(t,l]}} \right) \\
& \stackrel{(47),(49)}{=} \alpha_{(S \cup \{i\})^{\tau_q}(q,l]} p_m - \alpha_m \left(p_{(S \cup \{i\})^{\tau_q}(q,l]} + p_{\overline{(S \cup \{i\})^{\tau_q}(q,l]}} + p_{\overline{(S \cup \{i\})^{\tau_m}(t,l]} \setminus (S \cup \{i\})^{\tau_q}(q,l]}} \right) \\
& \quad + \alpha_q p_m - \alpha_m p_q \\
& \stackrel{u_q < u_m}{<} \alpha_{(S \cup \{i\})^{\tau_q}(q,l]} p_m - \alpha_m \left(p_{(S \cup \{i\})^{\tau_q}(q,l]} + p_{\overline{(S \cup \{i\})^{\tau_q}(q,l]}} \right) \\
& = \alpha_{(S \cup \{i\})^{\tau_q}(q,l]} p_m - \alpha_m p_{N^{\tau_q}(q,l]} \stackrel{(42)}{<} 0,
\end{aligned}$$

if player q has not been moved by the algorithm. Hence, by using the additive structure of cost differences we have shown that moving player m behind player l in τ_m is not beneficial.

C On assumption 2 for Theorem 4.1 in Section 4

We will prove that without loss of generality we can assume

$$(S \cup \{j\} \cup \{i\})_{c(j, S \cup \{j\} \cup \{i\}, \sigma_0)}^{\sigma_0} = \{j\},$$

and

$$(S \cup \{j\} \cup \{i\})_{c(i, S \cup \{j\} \cup \{i\}, \sigma_0)}^{\sigma_0} = \{i\},$$

in order to prove the convexity of SoSi sequencing games. Suppose that player i is not the only player in his component in σ_0 , i.e.,

$$\{i\} \subsetneq (S \cup \{j\} \cup \{i\})_{c(i, \sigma_0)}^{\sigma_0}.$$

Then, for example, the direct predecessor of player i in σ_0 is a member of $S \cup \{j\} \cup \{i\}$. We can define a different one-machine sequencing situation $(\bar{N}, \bar{\sigma}_0, \bar{p}, \bar{\alpha})$ where $\bar{N} = N \cup \{d\}$ with $d \notin N$,

$$\begin{aligned}
\bar{\sigma}_0(k) &= \begin{cases} \sigma_0(k) & \text{if } k \in P(\sigma_0, i), \\ \sigma_0(i) & \text{if } k = d, \\ \sigma_0(k) + 1 & \text{if } k \in \{i\} \cup F(\sigma_0, i), \end{cases} \\
\bar{p}_k &= \begin{cases} p_k & \text{if } k \in N, \\ 0 & \text{if } k = d, \end{cases}
\end{aligned}$$

and

$$\bar{\alpha}_k = \begin{cases} \alpha_k & \text{if } k \in N, \\ 0 & \text{if } k = d. \end{cases}$$

Hence, this new sequencing situation $(\bar{N}, \bar{\sigma}_0, \bar{p}, \bar{\alpha})$ is obtained from the original sequencing situation (N, σ_0, p, α) by adding a dummy player, with processing time and costs per time unit both equal to zero, directly in front of player i such that the predecessor of player i does not belong to $S \cup \{j\} \cup \{i\}$ anymore.

Let (\bar{N}, \bar{v}) be the SoSi sequencing game corresponding to one-machine sequencing situation $(\bar{N}, \bar{\sigma}_0, \bar{p}, \bar{\alpha})$. Although $\alpha_d = 0$ and $p_d = 0$, we can still apply the algorithm with respect to coalition $S \cup \{j\} \cup \{i\}$ and initial processing order $\bar{\sigma}_0$, because player d is not a member of $S \cup \{j\} \cup \{i\}$. The mutual order of the players in N will be the same in $\text{Alg}((N, \sigma_0, p, \alpha), S \cup \{j\} \cup \{i\})$ and $\text{Alg}((\bar{N}, \bar{\sigma}_0, \bar{p}, \bar{\alpha}), S \cup \{j\} \cup \{i\})$. Therefore,

$v(S \cup \{j\} \cup \{i\}) = \bar{v}(S \cup \{j\} \cup \{i\})$. Similarly, we have $v(S) = \bar{v}(S)$, $v(S \cup \{i\}) = \bar{v}(S \cup \{i\})$ and $v(S \cup \{j\}) = \bar{v}(S \cup \{j\})$.

Note that if also the direct follower of player i in σ_0 is a member of $S \cup \{j\} \cup \{i\}$, then we also add a dummy player directly behind player i . By adding a dummy player directly in front and behind player i , player i will be the only player in his component. Moreover, all arguments can also be applied to player j . Hence, without loss of generality we can assume that player i and j are both the only player in their component in σ_0 .

D Proof of claims 1 – 4 in Theorem 4.1

Claim 1: $[c(k, \sigma_{S \cup \{j\} \cup \{i\}}) = c(k, \sigma_{S \cup \{i\}}) \geq c(i, \sigma_S)]$ for all $k \in M^i(S)$

From (9) it follows that $c(k, \sigma_{S \cup \{i\}}) \geq c(i, \sigma_S)$ for all $k \in M^i(S)$. Moreover, from Proposition 3.5 it follows that if we go from $\sigma_{S \cup \{i\}}$ to $\sigma_{S \cup \{j\} \cup \{i\}}$, then the players to the right of player j in $\sigma_{S \cup \{i\}}$ will stay in the same component. Since all players in $M^i(S)$ are to the right of player j in $\sigma_{S \cup \{i\}}$, we have

$$c(k, \sigma_{S \cup \{j\} \cup \{i\}}) = c(k, \sigma_{S \cup \{i\}}),$$

for all $k \in M^i(S)$.

Claim 2: $[c(k, \sigma_{S \cup \{j\} \cup \{i\}}) = c(k, \sigma_{S \cup \{j\}})]$ for all $k \in M_{1c}^i(S)$

From Proposition 3.5 it follows that if we go from $\sigma_{S \cup \{j\}}$ to $\sigma_{S \cup \{j\} \cup \{i\}}$, then the players to the right of player i in $\sigma_{S \cup \{j\}}$ will stay in the same component. Since all players in $M_{1c}^i(S)$ are to the right of player i in $\sigma_{S \cup \{j\}}$, we have

$$c(k, \sigma_{S \cup \{j\} \cup \{i\}}) = c(k, \sigma_{S \cup \{j\}}),$$

for all $k \in M_{1c}^i(S)$.

Claim 3: $[c(k, \sigma_S) = c(k, \sigma_{S \cup \{j\}})]$ for all $k \in M_2^i(S)$

From Proposition 3.5 it follows that if we go from σ_S to $\sigma_{S \cup \{j\}}$, then the players to the right of player j in σ_S will stay in the same component. Since all players in $M_2^i(S)$ are to the right of player j in σ_S , we have

$$c(k, \sigma_S) = c(k, \sigma_{S \cup \{j\}}),$$

for all $k \in M_2^i(S)$.

Claim 4: $[c(k, \sigma_S) = c(k, \sigma_{S \cup \{j\}})]$ for all $k \in M_{1a}^i(S)$

From Proposition 3.6 it follows that the players who are in $\sigma_{S \cup \{j\}}$ to the left of the original component of player j have not been moved when going from σ_S to $\sigma_{S \cup \{j\}}$. Since all players in $M_{1a}^i(S)$ are to the left of the original component of player j in $\sigma_{S \cup \{j\}}$, we have

$$c(k, \sigma_S) = c(k, \sigma_{S \cup \{j\}}),$$

for all $k \in M_{1a}^i(S)$.

E Proof of claim 5 in Theorem 4.1

Let $k \in M_{1a}^i(S) \cup M_{1b}^i(S) \cup M_2^i(S)$, we will prove

$$\bar{\delta}_k \leq \delta_k.$$

Note that we consider the players in $M_{1a}^i(S) \cup M_{1b}^i(S) \cup M_2^i(S)$ from the right to the left with respect to $\bar{\sigma}$ and $\sigma_{S \cup \{j\}}$. So, if $i \in M_{1a}^i(S) \cup M_{1b}^i(S) \cup M_2^i(S)$, then player i is the first player who is moved. From now on we distinguish between two cases: $k = i$ and $k \neq i$.

Case 1: $[k = i]$

As player i is the first player who is moved, we have $\bar{\tau}_i = \bar{\sigma}$ and $\tau_i = \sigma_{S \cup \{j\}}$. Note that we have

$$\bar{\delta}_i = \alpha_{(S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} p_i - \alpha_i p_{N^{\bar{\tau}_i}(i, \bar{r}_i]},$$

and

$$\delta_i = \alpha_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i]} p_i - \alpha_i p_{N^{\tau_i}(i, r_i]}.$$

In order to prove $\bar{\delta}_i \leq \delta_i$, we compare the two sets of players that player i jumps over in $\bar{\tau}_i$ and τ_i . We will show that all players that player i jumps over in $\bar{\tau}_i$, player i also jumps over in τ_i . However, there might be some players that player i jumps over in τ_i but not in $\bar{\tau}_i$. It can be shown that these extra players that player i jumps over in τ_i all have a higher urgency than player i and thus this results in extra cost savings. Formally, we show the following three statements:

- Statement 1(a): $(\overline{S \cup \{i\}})^{\bar{\tau}_i}(i, \bar{r}_i] = (\overline{S \cup \{j\} \cup \{i\}})^{\tau_i}(i, r_i]$,
- Statement 1(b): $(S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i] \subseteq (S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i]$,
- Statement 1(c): if $(S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i] \subsetneq (S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i]$, then

$$\alpha_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \setminus (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} p_i - \alpha_i p_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \setminus (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} > 0.$$

Proof of statement 1(a). From claim 1 it follows that $c(i, \sigma_{S \cup \{i\}}) = c(i, \sigma_{S \cup \{j\} \cup \{i\}})$ and thus player i will be moved in both processing orders to the same component, i.e.,

$$c(\bar{\tau}_i, \bar{\tau}_i) = c(\tau_i, \tau_i).$$

Moreover, by the definition of $\bar{\sigma}$ and since $i \notin S \cup \{j\}$, we have $c(i, \bar{\sigma}) = c(i, \sigma_{S \cup \{j\}})$ and thus

$$c(i, \bar{\tau}_i) = c(i, \tau_i). \tag{50}$$

Hence, player i is moved in $\bar{\tau}_i$ and τ_i from the same component and to the same component, so player i jumps in $\bar{\tau}_i$ and τ_i over the same players outside $S \cup \{j\} \cup \{i\}$. Moreover, because $\bar{\sigma}(j) < \bar{\sigma}(i)$ and thus $\bar{\tau}_i(j) < \bar{\tau}_i(i)$, we have $j \notin N^{\bar{\tau}_i}(i, \bar{r}_i]$. To summarize,

$$(\overline{S \cup \{i\}})^{\bar{\tau}_i}(i, \bar{r}_i] = (\overline{S \cup \{j\} \cup \{i\}})^{\tau_i}(i, \bar{r}_i] = (\overline{S \cup \{j\} \cup \{i\}})^{\tau_i}(i, r_i].$$

Proof of statement 1(b). Let $l \in (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]$. From (12) we know that player l is in $\bar{\tau}_i$ in a component at most as far to the right as in τ_i , i.e.,

$$c(l, \bar{\tau}_i) \leq c(l, \tau_i).$$

Moreover, from $l \in (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]$ and (50) it follows that

$$c(i, \tau_i) = c(i, \bar{\tau}_i) < c(l, \bar{\tau}_i).$$

Note that there is a strict inequality because player i is the only player in his component. Combining the previous two equations we get

$$c(i, \tau_i) < c(l, \tau_i), \quad (51)$$

i.e., player l is also to the right of player i in τ_i . Now suppose player l is to the right of player r_i in τ_i , then player l is to the right of player i in $\sigma_{S \cup \{j\} \cup \{i\}}$ and thus

$$c(l, \sigma_{S \cup \{i\}}) \leq c(i, \sigma_{S \cup \{i\}}) = c(i, \sigma_{S \cup \{j\} \cup \{i\}}) \leq c(l, \sigma_{S \cup \{j\} \cup \{i\}}),$$

where the first inequality follows from $l \in (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]$ and the first equality follows from claim 1. Therefore, it follows that the swap of player i and player l with respect to $\sigma_{S \cup \{j\} \cup \{i\}}$ is admissible. From Proposition 3.1 it then follows that $u_i \geq u_l$. However, since player i jumps over player l when going from $\bar{\sigma}$ to $\sigma_{S \cup \{i\}}$, it follows from Proposition 3.1 that $u_l > u_i$, which contradicts $u_i \geq u_l$. Therefore l cannot be to the right of player r_i in τ_i , so player l is to the left of player r_i in τ_i . Combining this result with (51) we have

$$l \in (S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i].$$

Proof of statement 1(c). Let $(S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i] \subsetneq (S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i]$. For every player $l \in (S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \setminus (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]$ we have $u_l > u_i$ since player i jumps over player l when going from $\sigma_{S \cup \{j\}}$ to $\sigma_{S \cup \{j\} \cup \{i\}}$ (cf. Proposition 3.1). Combining this with Proposition 3.2 applied on the set $U = (S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \setminus (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]$ and player i , we have that

$$\frac{\alpha_i}{p_i} < \frac{\alpha_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \setminus (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]}}{p_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \setminus (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]}},$$

i.e., $\alpha_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \setminus (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} p_i - \alpha_i p_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \setminus (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} > 0$.

Note that if in statement 1(b) we have equality, then the inequality $\bar{\delta}_i \leq \delta_i$ follows immediately from statement 1(a). Next, if in statement 1(b) we have a strict subset, then

$$\begin{aligned} \bar{\delta}_i &= \alpha_{(S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} p_i - \alpha_i p_{N^{\bar{\tau}_i}(i, \bar{r}_i]} \\ &= \alpha_{(S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} p_i - \alpha_i \left(p_{(S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} + p_{\overline{(S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i)}} \right) \\ &\stackrel{1(c)}{<} \alpha_{(S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} p_i - \alpha_i \left(p_{(S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} + p_{\overline{(S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i)}} \right) \\ &\quad + \alpha_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \setminus (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} p_i - \alpha_i p_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \setminus (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} \\ &\stackrel{1(a)}{=} \left(\alpha_{(S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} + \alpha_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \setminus (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} \right) p_i \\ &\quad - \alpha_i \left(p_{(S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} + p_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \setminus (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i]} + p_{\overline{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i] \setminus (S \cup \{i\})^{\bar{\tau}_i}(i, \bar{r}_i)}} \right) \\ &= \alpha_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i]} p_i - \alpha_i \left(p_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i]} + p_{\overline{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i]}} \right) \\ &= \alpha_{(S \cup \{j\} \cup \{i\})^{\tau_i}(i, r_i]} p_i - \alpha_i p_{N^{\tau_i}(i, r_i]} \\ &= \delta_i. \end{aligned}$$

The idea behind the previous strict inequality is as follows. Player i jumps in $\bar{\tau}_i$ over the same players as in τ_i , but additionally player i jumps in τ_i also over some extra players.

It follows from statement 1(c) that these extra players all have a higher urgency and thus the jump of player i over those extra players results in cost savings.

Case 2: $[k \neq i]$

The difference with respect to case 1 is that in this case player k is not necessarily the only player in its component, while in case 1 player i was the only player in its component of $S \cup \{j\} \cup \{i\}$ with respect to σ_0 . Another difference is that now player k might not be the first player who is moved, and thus $\bar{\tau}_k$ and $\bar{\sigma}$, and τ_i and $\sigma_{S \cup \{j\}}$ might differ.

Note that

$$\bar{\delta}_k = \alpha_{(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{\tau}_k]} p_k - \alpha_k p_{N^{\bar{\tau}_k}(k, \bar{\tau}_k]},$$

and

$$\delta_k = \alpha_{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]} p_k - \alpha_k p_{N^{\tau_k}(k, r_k]}.$$

In order to prove $\bar{\delta}_k \leq \delta_k$, we compare the two sets of players that player k jumps over in $\bar{\tau}_k$ and τ_k . We will show that all players, excluding player j , that player k jumps over in $\bar{\tau}_k$, player k also jumps over in τ_k . Formally, we show the following three statements (which are similar to statements 1(a)-(c)):

- Statement 2(a): $\left(\overline{(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{\tau}_k]} \right) \setminus \{j\} = \overline{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]}$,
- Statement 2(b): $(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{\tau}_k] \subseteq ((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]) \setminus \{j\}$,
- Statement 2(c): if $(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{\tau}_k] \subsetneq ((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]) \setminus \{j\}$, then

$$\alpha_{(((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]) \setminus \{j\}) \setminus ((S \cup \{i\})^{\bar{\tau}_k}(k, \bar{\tau}_k])} p_k - \alpha_k p_{(((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]) \setminus \{j\}) \setminus ((S \cup \{i\})^{\bar{\tau}_k}(k, \bar{\tau}_k])} > 0.$$

Note that in the proof of statement 1(a) and 1(c) the fact that player i is the only player in its component is not used and therefore the proofs for statements 2(a) and 2(b) are similar.

Proof of statement 2(b). Let $l \in (S \cup \{i\})^{\bar{\tau}_k}(k, \bar{\tau}_k]$. From (13) we know that player l is in $\bar{\tau}_k$ in a component at most as far to the right as in τ_k , i.e.,

$$c(l, \bar{\tau}_k) \leq c(l, \tau_k).$$

Moreover, from (11) we have $c(k, \bar{\sigma}) = c(k, \sigma_{S \cup \{j\}})$ and thus $c(k, \bar{\tau}_k) = c(k, \tau_k)$. Together with $l \in (S \cup \{i\})^{\bar{\tau}_k}(k, \bar{\tau}_k]$ it follows that

$$c(k, \tau_k) = c(k, \bar{\tau}_k) \leq c(l, \bar{\tau}_k).$$

Combining the previous two equations we get

$$c(k, \tau_k) \leq c(l, \tau_k). \tag{52}$$

This implies

$$\tau_k(k) < \tau_k(l).$$

For this, note that if we have an equality in (52), then $c(k, \bar{\tau}_k) = c(l, \bar{\tau}_k)$. Moreover, since $l \in (S \cup \{i\})^{\bar{\tau}_k}(k, \bar{\tau}_k]$, we know $\bar{\tau}_k(k) < \bar{\tau}_k(l)$ and thus also $\tau_k(k) < \tau_k(l)$ (because both τ_k and $\bar{\tau}_k$ are urgency respecting processing orders and moreover because the tiebreaking

rule, in case of equal urgencies, mentioned in condition (iii) of $\bar{\sigma}$ is the same tiebreaking rule as in property (iv) of the algorithm). On the other hand, if there is a strict inequality in (52), then automatically $\tau_k(k) < \tau_k(l)$. Using similar arguments as in the proof of statement 1(b) we have

$$l \in N^{\tau_k}(k, r_k],$$

and since $l \in S \cup \{i\}$, we have

$$l \in ((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]) \setminus \{j\}.$$

Now we continue with the main line of the proof. Using similar arguments as in case 1, it follows from statement 2(a)-(c) that

$$\begin{aligned} & \alpha_{(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{r}_k]} p_k - \alpha_k \left(p_{(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{r}_k]} + p_{((\overline{S \cup \{i\}})^{\bar{\tau}_k}(k, \bar{r}_k]) \setminus \{j\}} \right) \\ \leq & \alpha_{((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]) \setminus \{j\}} p_k - \alpha_k \left(p_{((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]) \setminus \{j\}} + p_{(\overline{S \cup \{j\} \cup \{i\}})^{\tau_k}(k, r_k]} \right). \end{aligned} \quad (53)$$

We distinguish from now on between the following four cases: case 2(i) where player k does not jump over player j in both $\bar{\tau}_k$ and τ_k , case 2(ii) where player k jumps over player j in both $\bar{\tau}_k$ and τ_k , case 2(iii) where player k jumps over player j in $\bar{\tau}_k$ but not in τ_k and case 2(iv) where player k jumps over player j in τ_k but not in $\bar{\tau}_k$.

Case 2(i): $[j \notin N^{\bar{\tau}_k}(k, \bar{r}_k] \text{ and } j \notin N^{\tau_k}(k, r_k)]$

Note that if in statement 2(b) we have equality, then the inequality $\bar{\delta}_k \leq \delta_k$ follows immediately from statement 2(a). Next, if in statement 2(b) we have a strict subset, then

$$\begin{aligned} \bar{\delta}_k &= \alpha_{(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{r}_k]} p_k - \alpha_k p_{N^{\bar{\tau}_k}(k, \bar{r}_k]} \\ &\stackrel{j \notin N^{\bar{\tau}_k}(k, \bar{r}_k]}{=} \alpha_{(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{r}_k]} p_k - \alpha_k \left(p_{(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{r}_k]} + p_{((\overline{S \cup \{i\}})^{\bar{\tau}_k}(k, \bar{r}_k]) \setminus \{j\}} \right) \\ &\stackrel{(53)}{<} \alpha_{((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]) \setminus \{j\}} p_k - \alpha_k \left(p_{((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]) \setminus \{j\}} + p_{(\overline{S \cup \{j\} \cup \{i\}})^{\tau_k}(k, r_k]} \right) \\ &\stackrel{j \notin N^{\tau_k}(k, r_k]}{=} \alpha_{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]} p_k - \alpha_k \left(p_{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]} + p_{(\overline{S \cup \{j\} \cup \{i\}})^{\tau_k}(k, r_k]} \right) \\ &= \alpha_{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]} p_k - \alpha_k p_{N^{\tau_k}(k, r_k]} \\ &= \delta_k. \end{aligned}$$

The idea behind the strict inequality is as follows. Player k jumps in $\bar{\tau}_k$ over the same players as in τ_k , but additionally player k jumps in τ_k also over some extra players. It follows from statement 2(c) that these extra players all have a higher urgency and thus the jump of player k over those extra players results in cost savings.

Case 2(ii): $[j \in N^{\bar{\tau}_k}(k, \bar{r}_k] \text{ and } j \in N^{\tau_k}(k, r_k)]$

It follows that

$$\begin{aligned} \bar{\delta}_k &= \alpha_{(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{r}_k]} p_k - \alpha_k p_{N^{\bar{\tau}_k}(k, \bar{r}_k]} \\ &\stackrel{j \in N^{\bar{\tau}_k}(k, \bar{r}_k]}{=} \alpha_{(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{r}_k]} p_k - \alpha_k \left(p_{(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{r}_k]} + p_{((\overline{S \cup \{i\}})^{\bar{\tau}_k}(k, \bar{r}_k]) \setminus \{j\}} + p_j \right) \\ &\stackrel{(53)}{\leq} \alpha_{((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]) \setminus \{j\}} p_k - \alpha_k \left(p_{((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k]) \setminus \{j\}} + p_{(\overline{S \cup \{j\} \cup \{i\}})^{\tau_k}(k, r_k]} + p_j \right) \end{aligned}$$

$$\begin{aligned}
& \stackrel{j \in N^{\tau_k}(k, r_k]}{=} \alpha_{((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k)) \setminus \{j\}} p_k - \alpha_k \left(p_{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k)} + p_{(\overline{S \cup \{j\} \cup \{i\}})^{\tau_k}(k, r_k)} \right) \\
& \stackrel{\alpha_j p_k > 0}{<} \left(\alpha_{((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k)) \setminus \{j\}} + \alpha_j \right) p_k - \alpha_k \left(p_{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k)} + p_{(\overline{S \cup \{j\} \cup \{i\}})^{\tau_k}(k, r_k)} \right) \\
& \stackrel{j \in N^{\tau_k}(k, r_k]}{=} \alpha_{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k)} p_k - \alpha_k \left(p_{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k)} + p_{(\overline{S \cup \{j\} \cup \{i\}})^{\tau_k}(k, r_k)} \right) \\
& = \alpha_{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k)} p_k - \alpha_k p_{N^{\tau_k}(k, r_k)} \\
& = \delta_k.
\end{aligned}$$

The idea behind the strict inequality is as follows. All players, including player j , that player k jumps over in $\bar{\tau}_k$, player k also jumps over in τ_k . However, as player j belongs to coalition $S \cup \{j\} \cup \{i\}$ and not to coalition $S \cup \{i\}$, there are some extra cost savings in δ_k . These extra cost savings are due to the reduction of the processing time for player j due to the jump of player k , namely $\alpha_j p_k$.

Case 2(iii): $[j \in N^{\bar{\tau}_k}(k, \bar{r}_k] \text{ and } j \notin N^{\tau_k}(k, r_k)]$

It follows that

$$\begin{aligned}
\bar{\delta}_k & = \alpha_{(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{r}_k)} p_k - \alpha_k p_{N^{\bar{\tau}_k}(k, \bar{r}_k)} \\
& \stackrel{j \in N^{\bar{\tau}_k}(k, \bar{r}_k]}{=} \alpha_{(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{r}_k)} p_k - \alpha_k \left(p_{(S \cup \{i\})^{\bar{\tau}_k}(k, \bar{r}_k)} + p_{((\overline{S \cup \{i\}})^{\bar{\tau}_k}(k, \bar{r}_k)) \setminus \{j\}} + p_j \right) \\
& \stackrel{(53)}{\leq} \alpha_{((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k)) \setminus \{j\}} p_k - \alpha_k \left(p_{((S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k)) \setminus \{j\}} + p_{(\overline{S \cup \{j\} \cup \{i\}})^{\tau_k}(k, r_k)} + p_j \right) \\
& \stackrel{j \notin N^{\tau_k}(k, r_k]}{=} \alpha_{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k)} p_k - \alpha_k \left(p_{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k)} + p_{(\overline{S \cup \{j\} \cup \{i\}})^{\tau_k}(k, r_k)} + p_j \right) \\
& \stackrel{\alpha_k p_j > 0}{<} \alpha_{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k)} p_k - \alpha_k \left(p_{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k)} + p_{(\overline{S \cup \{j\} \cup \{i\}})^{\tau_k}(k, r_k)} \right) \\
& = \alpha_{(S \cup \{j\} \cup \{i\})^{\tau_k}(k, r_k)} p_k - \alpha_k p_{N^{\tau_k}(k, r_k)} \\
& = \delta_k.
\end{aligned}$$

The idea behind the strict inequality is as follows. All players, excluding player j , that player k jumps over in $\bar{\tau}_k$, player k also jumps over in τ_k . However, as player k jumps over player j in $\bar{\tau}_k$ and not in τ_k , the completion time of player k will increase with at least p_j more in τ_k than in $\bar{\tau}_k$ and thus the cost savings in $\bar{\delta}_k$ are less than in δ_k .

Case 2(iv): $[j \notin N^{\bar{\tau}_k}(k, \bar{r}_k] \text{ and } j \in N^{\tau_k}(k, r_k)]$

We will show that this case is not possible. As player k jumps over player j when going from $\sigma_{S \cup \{j\}}$ to $\sigma_{S \cup \{j\} \cup \{i\}}$, player j did not jump over player k when going from σ_S to $\sigma_{S \cup \{j\}}$ (otherwise there would be a contradiction with respect to the urgencies, cf. Proposition 3.1). Hence, the mutual order of player k and player j is in σ_S the same as in $\sigma_{S \cup \{j\}}$ and thus $\sigma_S(k) < \sigma_S(j)$. Using Figure 10 this implies that $k \in M_1^i(S)$. Moreover, as $j \notin N^{\bar{\tau}_k}(k, \bar{r}_k]$ it follows from Figure 12 that $k \notin M_{1a}^i(S)$ and thus $k \in M_{1b}^i(S)$. Therefore, it follows from Figure 10(c) that $c(k, \sigma_{S \cup \{j\}}) \geq c(j, \sigma_S)$. Hence,

$$c(j, \sigma_S) \leq c(k, \sigma_{S \cup \{j\}}) \leq c(j, \sigma_{S \cup \{j\}}),$$

where the last inequality follows from $j \in N^{\tau_k}(k, r_k]$. Therefore, it follows that the swap of player k and player j with respect to $\sigma_{S \cup \{j\}}$ is admissible.

From Proposition 3.1 it then follows that $u_k \geq u_j$. However, since player k jumps over player j when going from $\sigma_{S \cup \{j\}}$ to $\sigma_{S \cup \{j\} \cup \{i\}}$, it follows from Proposition 3.1 that $u_j > u_k$ too, a contradiction.